

## Раздел IV. Облачные вычисления

УДК 004.023, 004.75

DOI 10.18522/2311-3103-2016-12-92103

**Н.А. Балашов, А.В. Баранов, И.С. Кадочников, В.В. Кореньков,  
Н.А. Кутовский, А.В. Нечаевский, И.С. Пелеванюк**

**ПРОГРАММНЫЙ КОМПЛЕКС ИНТЕЛЛЕКТУАЛЬНОГО  
ДИСПЕТЧИРОВАНИЯ И АДАПТИВНОЙ САМООРГАНИЗАЦИИ  
ВИРТУАЛЬНЫХ ВЫЧИСЛИТЕЛЬНЫХ РЕСУРСОВ НА БАЗЕ  
ОБЛАЧНОГО ЦЕНТРА ЛИТ ОИЯИ\***

*В Лаборатории информационных технологий Объединенного института ядерных исследований (ЛИТ ОИЯИ) несколько лет назад была развернута и по сей день успешно функционирующая облачная инфраструктура, построенная по модели Инфраструктура как услуга (Infrastructure as a Service, IaaS) на платформе с открытым исходным кодом OpenNebula. Детальный анализ накопленного на протяжении нескольких лет опыта использования и управления облачной инфраструктурой позволил выявить в ее работе некоторые слабые места. Наиболее актуальной проблемой на сегодняшний день является неэффективное использование имеющихся в облаке ресурсов являющееся следствием использования облачных ресурсов для решения большого количества разнородных задач и используемых программных средств. По результатам анализа, проведенного группой специалистов ЛИТ ОИЯИ, была показана необходимость оптимизации облачной инфраструктуры ОИЯИ. В качестве решения данной проблемы был предложен метод интеллектуального диспетчирования ресурсов. В статье приведен подробный анализ комплекса проблем, приводящих к неэффективному использованию вычислительных ресурсов. В качестве решения описанных проблем представлен метод интеллектуального диспетчирования ресурсов, который позволит высвободить некоторое количество ресурсов за счет динамического перераспределения и уплотнения пользовательских виртуальных ресурсов. Использование технологий уплотнения виртуальных ресурсов неизбежно ведет к снижению качества обслуживания, для контроля которого предлагается использование системы ранжирования вычислительных ресурсов, которая позволит поддерживать качество обслуживания пользователей на заданном уровне. Высвобожденные в результате применения предлагаемого метода ресурсы в целях повышения эффективности могут быть повторно использованы. В статье предложено несколько стратегий повторного использования высвобожденных ресурсов, в том числе за счет интеграции с системами пакетной обработки заданий. Следование данным стратегиям позволит существенно повысить эффективность использования ресурсов, составляющих облачную инфраструктуру. Также рассмотрен программный комплекс интеллектуального диспетчирования с открытым исходным кодом и модульной архитектурой, разрабатываемый на основе предложенных методов и стратегий.*

*Облачные вычисления; виртуализация; оптимизация; интеллектуальное управление; ЦОД; консолидация VM.*

\* Работа выполнена при поддержке РФФИ, проект № 15-29-07027.

**N.A. Balashov, A.V. Baranov, I.S. Kadochnikov, V.V. Korenkov, N.A. Kutovskiy,  
A.V. Nechaevskiy, I.S. Pelevanyuk**

**SOFTWARE COMPLEX FOR INTELLIGENT SCHEDULING AND ADAPTIVE  
SELF-ORGANIZATION OF VIRTUAL COMPUTING RESOURCES BASED  
IN LIT JINR CLOUD CENTER**

*Rapid development of cloud technologies has led to their wide use both in commercial and academic areas. The Joint Institute for Nuclear Research (JINR) hosts its own cloud infrastructure built on the OpenNebula platform based on the Infrastructure as a Service (IaaS) model. The JINR cloud service is used as a universal computing resource supporting workload-intensive scientific computations as well as low-load activities. The complexity of modern software libraries and applications makes it hard to predict possible workloads generated by the software. For this reason cloud resources are often over-allocated leading to a high degree of underutilization of underlying equipment. It is useful for commercial cloud providers to optimize their resources consumption but it is especially important for scientific organizations, such as JINR, limited in resources, where it is crucial to get the maximum performance out of the owned computing infrastructure. A common solution is to use consolidation of virtual machines (VMs) and a number of algorithms and methods for dynamic reallocation and consolidation of VMs were proposed recently. In this paper we describe a new heuristics-based method of dynamic VMs reallocation utilizing the ranking system of computing resources to control Quality of Service (QoS) while over-provisioning the cloud infrastructure. The method allows to release an amount of computing capacities by the means of VMs consolidation and a set of approaches to deal with the released resources is described in the paper giving the detailed overview of a novel approach based on integration of cloud services with batch systems. Based on the proposed methods and strategies we review a universal software framework that is currently under development allowing implementing a complete solution for improving cloud environments built on IaaS model.*

*Cloud computing; virtualization; optimization; intelligent control; datacenters; VM consolidation.*

**Введение.** Современные облачные вычислительные среды (ОВС) характеризуются высокой гибкостью и масштабируемостью, а в некоторых случаях – и универсальностью, что в конечном итоге привело к их широкому распространению в коммерческой среде. В научной среде облачные технологии продолжают набирать популярность и применяются как при построении сложных систем распределенных вычислений для обработки больших объемов данных [1–3], так и для поддержки работы информационных систем (ИС) и предоставления универсального вычислительного ресурса конечным пользователям [3], который является для них наиболее простым способом получения вычислительных ресурсов, необходимых для решения их повседневных задач.

Примером одной из подобных научных облачных вычислительных сред является облачный сервис Объединенного института ядерных исследований (ОИЯИ). Данный сервис построен на платформе OpenNebula по модели Инфраструктура как услуга (Infrastructure as a Service, IaaS) и используется как универсальная вычислительная инфраструктура, которая позволяет решать широкий спектр как научных, так и сервисных задач организации [3].

Облачный сервис ОИЯИ в настоящий момент активно используется пользователями, а процесс и результаты его использования постоянно анализируются. Анализ накопленного за последние несколько лет опыта работы с облачной вычислительной средой позволил сделать вывод о том, что входящие в состав облачного сервиса ОИЯИ вычислительные ресурсы используются недостаточно эффективно. Результаты проведенного анализа стали основой для разработки методов и стратегий повышения эффективности использования ресурсов.

**1. Описание проблемы и причин ее возникновения (анализ использования облачной вычислительной среды ОИЯИ).** Сложность современных ИС и приложений, включающих разнообразные задачи от интенсивных долговременных научных симуляций до транзакционной обработки запросов, делает задачу оценки необходимых ресурсов практически неразрешимой, поэтому при запросе вычислительных ресурсов все пользователи придерживаются единой стратегии: запросить как можно больше ресурсов. Следование данной стратегии неизбежно ведет к неэффективному расходованию вычислительных ресурсов с точки зрения тех, кто эти ресурсы предоставляет.

Облачный сервис ОИЯИ является универсальным вычислительным ресурсом, пользователей которого условно можно разделить на 4 группы:

- ◆ разработчики программного обеспечения (ПО);
- ◆ системные администраторы;
- ◆ физики;
- ◆ автоматизированные системы.

Профиль вычислительных нагрузок в виртуальных машинах (ВМ) разработчиков ПО характеризуется достаточно большим потреблением ресурсов в течение короткого промежутка времени. Как правило, это объясняется настройкой системного окружения, установкой дополнительного ПО и библиотек, необходимых для дальнейшей работы. Этот период обычно длится от одного до нескольких рабочих дней и напрямую зависит от сложности разрабатываемого проекта. Последующий период работы ВМ характеризуется малым потреблением ресурсов с краткосрочными повышениями нагрузок, иногда до пиковых значений. Повышенные нагрузки в этот период возникают в основном в моменты сборки, компиляции и тестирования разрабатываемого ПО. Частота возникновения повышенных нагрузок в этот период напрямую зависит от интенсивности разработки конкретного проекта. По окончании основного цикла разработки ПО, как правило, начинается период, когда частота появления нагрузок падает пропорционально спаду интенсивности ведения разработки проекта. Подобные виртуальные машины зачастую живут на протяжении всего цикла разработки и удаляются лишь по окончании разработки, либо при выходе владельца данной ВМ из команды разработки конкретного проекта.

Виртуальные машины системных администраторов используются, как правило, для изучения новых информационных систем и технологий, а также тестирования обновлений администрируемых ими информационных систем (ИС). Как и в случае с ВМ разработчиков ПО, ВМ для тестирования обновлений ИС характеризуются высоким потреблением ресурсов в начальный период работы ВМ, необходимостью первоначальной настройки системного окружения. Последующий период работы ВМ характеризуется редкими скачкообразными повышениями нагрузки в моменты выхода новых обновлений тестируемой ИС. Цикл жизни ВМ иногда может быть равен времени эксплуатации ИС.

ВМ первых двух групп пользователей имеют одно общее свойство – в нерабочее время нагрузка на ВМ значительно падает.

Профили ВМ третьей группы пользователей, физиков, характеризуются чередованием достаточно длительных временных интервалов (иногда до месяца) с пиковыми нагрузками и с близкими к минимальным (использование процессора может упасть до 0 %).

ВМ четвертой группы пользователей – это ВМ, создаваемые автоматически различными ИС и системами управления нагрузкой (workload management system, WMS) вычислительных инфраструктур различных физических экспериментов. На данный момент в облачном сервисе ОИЯИ это ВМ экспериментов ALICE, NOvA и BES-III.

Отдельно стоит упомянуть виртуальные машины, в которых развернуты информационные системы уже введенные в эксплуатацию, пользователями которых потенциально являются все сотрудники организации – нагрузка в подобных ВМ напрямую зависит от популярности ИС среди пользователей.

Не вызывает сомнений, что подобное использование ресурсов приводит к крайне неэффективному распределению вычислительных мощностей, а рис. 1 и 2 наглядно демонстрируют, что с ростом количества серверов облачного сервиса ОИЯИ проблема сохраняется. В тоже время на этих рисунках видно, что выделенное количество вычислительных ресурсов более чем в два раза превышает количество фактически используемых.

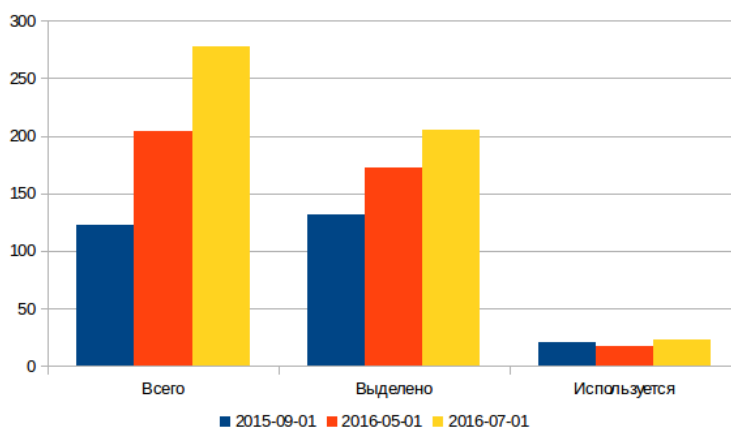


Рис. 1. Использование процессорных ресурсов облачного сервиса ОИЯИ, условные единицы

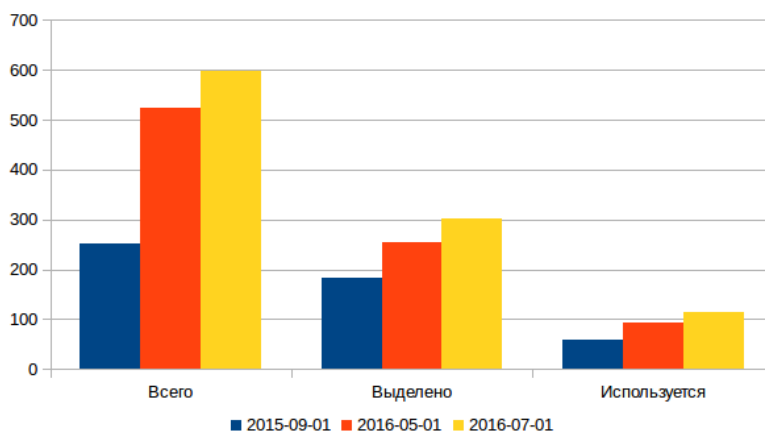


Рис. 2. Использование оперативной памяти облачного сервиса ОИЯИ, ГБ

## 2. Динамическое перераспределение и уплотнение виртуальных ресурсов.

Традиционным методом повышения эффективности использования ресурсов облачных вычислительных сред, построенных по модели IaaS, является уплотнение распределения виртуальных машин, использующее технологии overcommitment'a. Некоторые платформы (например, OpenStack), на которых строятся данные ОВС,

способны осуществлять подобное уплотнение виртуальных машин в соответствии с простейшими алгоритмами, в то время как в других платформах (например, OpenNebula) подобные функциональные возможности отсутствуют. Тем не менее, использование *overcommitment*'а неизбежно ведет к снижению уровня качества обслуживания (Quality of Service, QoS), так как никогда нельзя заранее точно предсказать будущую нагрузку, и возможно возникновение ситуаций, в которых реальная генерируемая нагрузка превысит максимально доступную физически из-за резкого роста нагрузки, генерируемой VM. Единственным решением проблемы снижения QoS на данный момент является постоянное отслеживание показателей производительности всех VM, запущенных в ОВС, и перераспределение VM между серверами, основанное на текущих показателях и необходимом для предотвращения перегрузок оборудования.

Современные технологии виртуализации позволяют производить «живую» миграцию виртуальных машин между различными серверами, то есть перенос VM с одного сервера на другой с минимальным временем ее простоя и полным восстановлением состояния запущенных процессов VM на предыдущем сервере.

Повышение эффективности использования оборудования ОВС на сегодняшний день является актуальной задачей, особенно в коммерческой среде. Это доказывает большое количество исследований, предлагающих различные методы динамической оптимизации виртуальных ресурсов ОВС [4–9]. Суть всех подобных методов сводится к оптимизации распределения виртуальных ресурсов на физическом оборудовании, организованного таким образом, чтобы максимально уплотнить распределение VM на серверах и при этом минимизировать потери QoS.

**3. Эвристический метод динамического перераспределения VM.** Для решения описанной выше проблемы неэффективного использования ресурсов облачного сервиса ОИЯИ авторами статьи был предложен метод динамического перераспределения виртуальных машин в облачной среде, основанный на анализе исторической информации о потребляемых виртуальными машинами ресурсах [10]. Предлагаемый метод позволяет высвободить часть используемых вычислительных ресурсов за счет более плотной компоновки виртуальных машин, используя технологии *overcommitment*'а. Алгоритм работы метода представлен на схеме на рис. 3.

Использование данного метода неизбежно приведет к снижению показателей надежности работы виртуальных машин, которая обеспечивалась выделением заведомо большего, чем реально имеется, количества виртуальных ресурсов и, соответственно, снижению QoS сервиса в целом. В предложенном методе используется система ранжирования реальных и виртуальных вычислительных ресурсов, которая является в том числе и инструментом регулирования степени надежности работы отдельных вычислительных ресурсов. Начальные ранги физических серверов задаются на основе экспертных оценок, которые в дальнейшем могут быть скорректированы в процессе работы системы.

**4. Повторное использование ресурсов.** В результате реализации предложенного метода может быть высвобождено некоторое количество ресурсов, и возникнет вопрос их повторного использования.

В качестве одного из возможных вариантов использования высвобожденных в результате применения данного метода ресурсов предлагается их возвращение обратно в пул (от англ. «pool») свободных ресурсов, повторная нагрузка на которые будет автоматически распределена стандартным облачным планировщиком, встроенным в используемую облачную платформу, в соответствии с заложенными в него алгоритмами.

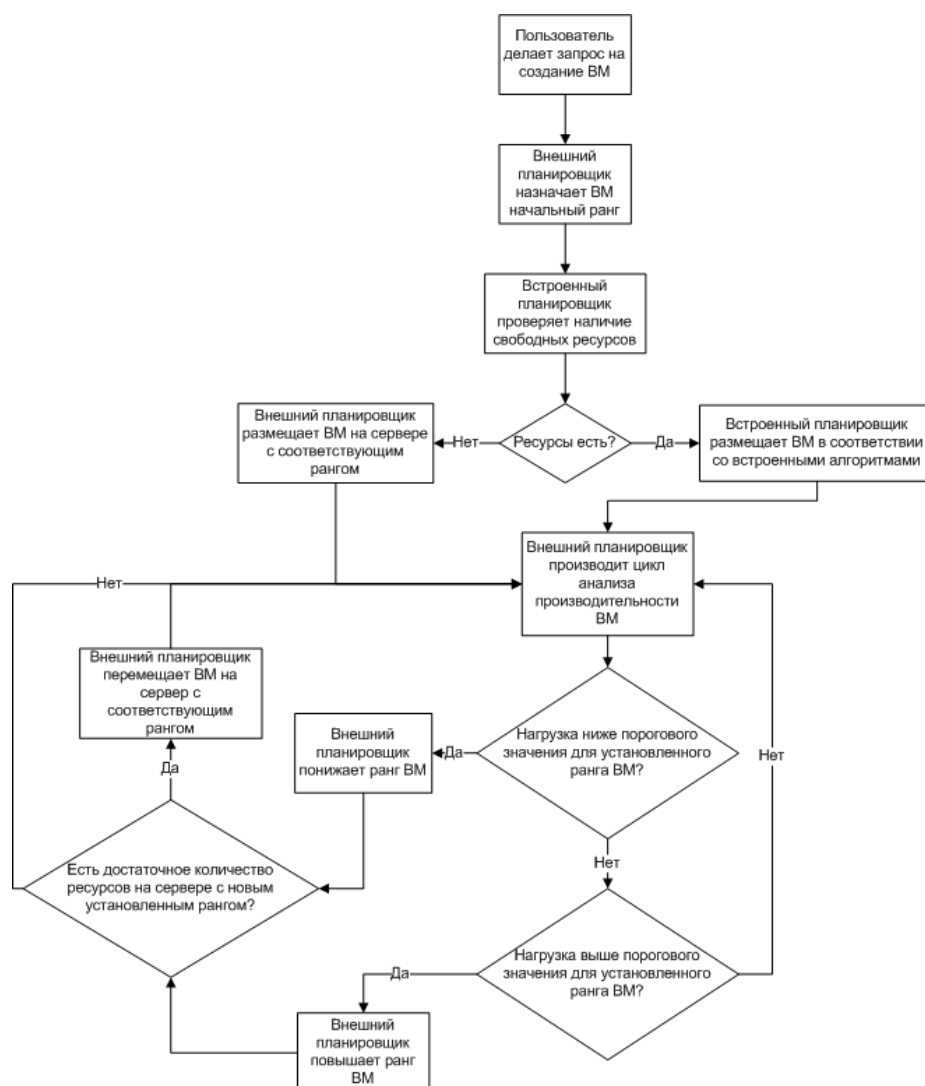


Рис. 3. Схема работы алгоритма динамического перераспределения ресурсов

В качестве альтернативного варианта может рассматриваться перевод высвобожденных ресурсов (серверов) в режим пониженного энергопотребления.

Другой альтернативой может быть повторное использование высвобожденных ресурсов научных облачных сред (подобных облачному сервису ОИЯИ) путем интеграции с системами пакетной обработки данных, что в конечном итоге может позволить минимизировать снижение надежности работы виртуальных машин при повышенной плотности их распределения на серверах. Некоторые системы подобного рода уже имеют возможность запуска вычислительных задач в облачные сервисы, построенные по модели IaaS: например, система управления распределенными ресурсами DIRAC [11]. Также ведутся разработки вспомогательных систем, дополняющих существующие системы подобным функционалом (VCondor [12], Vcycle [13]).

В данный момент в ОИЯИ ведется создание виртуального вычислительного кластера на базе системы HTCondor (необходимого для обработки вычислительных задач эксперимента NOvA), который может использовать свободные ресурсы облачного сервиса ОИЯИ.

HTCondor является системой пакетной обработки пользовательских задач, которые выполняются на вычислительных узлах. В качестве вычислительного узла в этой системе может выступать виртуальная машина, поэтому нужным образом заранее сконфигурированная виртуальная машина облачной среды может в любой момент подключиться к системе HTCondor и стать вычислительным узлом. По умолчанию HTCondor не может запрашивать дополнительные вычислительные узлы в облачном сервисе, однако определенным образом сконфигурированная виртуальная машина при запуске может самостоятельно подключиться к вычислительной инфраструктуре, поэтому при наличии свободных ресурсов облачному планировщику достаточно запустить эту виртуальную машину и она автоматически станет вычислительным узлом инфраструктуры HTCondor. Для обратного освобождения ресурсов планировщик в любой момент может остановить необходимое количество виртуальных вычислительных узлов HTCondor.

Вышеописанная схема является простейшей схемой, не учитывающей возможное отсутствие задач в очередях HTCondor. В этом случае виртуальные вычислительные узлы HTCondor по-прежнему будут запускаться, приводя к нерациональному расходованию вычислительных мощностей. Реализация же функционала отслеживания состояния очередей HTCondor диспетчером облачных ресурсов приведет к потере его универсальности. Для решения этой проблемы предлагается ввести в алгоритм диспетчирования систему маркировки виртуальных машин, являющихся временными (в данном случае являющихся временными вычислительными узлами HTCondor), чтобы планировщик автоматически мог останавливать их в случае возникновения необходимости в высвобождении ресурсов, а механизмы запуска виртуальных машин и отслеживания очередей задач делегировать какой-либо сторонней системе (например, VCondor).

**5. Программная реализация.** В данный момент в ОИЯИ ведется разработка программного комплекса интеллектуального диспетчирования виртуальных ресурсов на базе вышеописанных методов и стратегий.

Несмотря на существование большого количества методов и алгоритмов динамического планирования виртуальных ресурсов, программных реализаций с открытым исходным кодом на момент написания статьи очень мало [15–17] и все они, во-первых, разработаны для использования только с определенными облачными платформами, и, во-вторых, в настоящий момент не развиваются.

Целью данного проекта является создание универсального программного фреймворка (от англ. «framework») максимально абстрагированного от конкретной реализации ОВС, предоставляющего только общие абстрактные программные интерфейсы взаимодействия с объектами ОВС, построенными по модели IaaS. При таком подходе становится возможной реализация алгоритмов диспетчирования виртуальных ресурсов независимо от разработки интерфейсов взаимодействия с конкретной ОВС. В качестве примера реализации такого взаимодействия с конкретной облачной платформой в рамках данного проекта разрабатывается интерфейс взаимодействия с платформой OpenNebula. Схема разрабатываемого фреймворка представлена на рис. 4.



Рис. 4. Схема работы фреймворка

**Ядро системы** является центральным связующим компонентом, объединяющим все остальные модули системы. Его ролью является координация работы всех остальных частей системы: модуля генерации распределения VM, модуля мониторинга и модулей взаимодействия с ОВС.

Важным компонентом системы является модуль мониторинга ОВС, так как именно на показателях производительности VM, собранных системой мониторинга, а в некоторых случаях – и накопленных исторических данных [5, 10], строятся все современные методы планирования и динамического перераспределения VM.

Средства мониторинга традиционных вычислительных кластеров и систем довольно хорошо развиты, и существует большое количество систем, хорошо решающих эту задачу. Однако все эти инструменты были спроектированы без учета особенностей современных ОВС и изначально предназначались для отслеживания работоспособности физического оборудования в статической (редко изменяемой) конфигурации, следовательно, не полностью удовлетворяют требованиям для отслеживания состояния виртуальных машин в рамках единой ОВС [18], количество которых, как и их распределение между физическими серверами, часто меняется. Также ситуацию усугубляет потенциально более стремительный рост вычислительных мощностей ОВС [19] – по общим оценкам центры обработки данных (ЦОД) современных коммерческих ОВС (например, Amazon EC2) могут содержать десятки тысяч серверов, способных поддерживать работу сотен тысяч виртуальных машин. Задача мониторинга и сбора статистики ОВС еще не получила общепринятого решения, и активно ведутся исследования новых подходов к решению данной задачи. Поэтому при разработке данного программного комплекса важно абстрагироваться от конкретной реализации системы мониторинга. Разрабатываемый **модуль мониторинга** реализует единый формат представления данных, которые используются алгоритмами планирования ресурсов, без привязки к конкретной реализации системы мониторинга и сбора статистики использования ОВС. Таким образом, разработка алгоритмов планирования может вестись независимо от реализации системы мониторинга.

**Модуль генерации распределения VM** реализует общие интерфейсы, необходимые для реализации алгоритмов планирования (получение информации о текущем распределении VM и о показателях за заданный временной интервал) и на основе этих данных генерирует оптимальное распределение VM.



**Модуль взаимодействия с ОВС** предназначен только для передачи управляющего воздействия ОВС (например, миграция ВМ с сервера 1 на сервер 2), тип и параметры которого задает модуль генерации распределения ВМ на основе реализованных и используемых алгоритмов. Взаимодействие с ОВС осуществляется с помощью программных интерфейсов (Application Programming Interface, API), предоставляемых подключенной облачной платформой. Все современные облачные платформы предоставляют программные интерфейсы управления, но на сегодняшний день не существует единого стандарта подобных интерфейсов (хотя попытки разработать стандарты предпринимаются, например, разрабатывается Open Cloud Computing Interface (ОСЦИ) [20]). По этой причине разрабатываемый в рамках данного проекта модуль взаимодействия с ОВС реализует определенный набор интерфейсов для осуществления взаимодействия с модулем генерации распределения ВМ, вызовы которых транслируются в API-вызовы конкретной ОВС, например, EC2 или ОСЦИ.

**Модуль обнаружения перегрузок**, так же как и база данных, необходимая для накопления статистики генерируемых нагрузок, является опциональным. Данный модуль реализован в виде агента, постоянно запущенного на каждом узле системы. Его задачей, в соответствии с названием, является постоянное наблюдение за работой всех ВМ на вычислительном узле, на котором он запущен, для оперативного обнаружения перегрузок и определения списка ВМ, которые необходимо мигрировать с данного узла для предотвращения сильных перегрузок. Данное решение является страховкой от резких скачков нагрузки в ВМ, так как централизованный анализ производительности всей системы может занимать продолжительное время, особенно при большом количестве запущенных в ней ВМ. При обнаружении перегрузки на подконтрольном узле, модуль обнаружения перегрузок сообщает модулю генерации распределения ВМ список виртуальных машин, которые требуется оперативно мигрировать с данного узла, который в свою очередь принимает решение о миграции этих ВМ на менее загруженные узлы.

**Заключение.** Современные облачные платформы широко распространены как в коммерческой, так и в научной среде. В случае ограниченности количества ресурсов, составляющих облачную инфраструктуру, неизбежно встает вопрос эффективности их использования. В качестве решения задачи повышения эффективности использования ресурсов облака предложено использование метода интеллектуального диспетчирования ресурсов, результатом применения которого является высвобождение ресурсов с целью их повторного использования, организованного одним из предложенных способов согласно выбранной стратегии, что позволит существенно повысить эффективность использования ресурсов облака.

#### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Helge Meinhard*. Virtualization, clouds and IaaS at CERN // VTDC '12 Proceedings of the 6th international workshop on Virtualization Technologies in Distributed Computing, ACM New York, NY, USA, 2012. – P. 27-28.
2. *Timm, S. et al.* Cloud Services for the Fermilab Scientific Stakeholders // J.Phys.Conf.Ser. – 2015. – Vol. 664, No. 2.
3. *Baranov A.V., Balashov N.A., Kutovskiy N.A., Semenov R.N.* JINR cloud infrastructure evolution // Physics of Particles and Nuclei Letters. – 2016. – Vol. 13, No. 5. – P. 672-675.
4. *Feller E., Rilling L., Morin C.* Energy-aware ant colony based workload placement in clouds // Proceedings of the 12th IEEE/ACM International Conference on Grid Computing, Lyon, France - 2011.
5. *Farahnakian F., Liljeberg P., Plosila J.* LiRCUP: Linear regression based CPU usage prediction algorithm for live migration of virtual machines in data centers // 39th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA). – 2013. – P. 357-364.

6. *Beloglazov A., Buyya R.* Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers // *Concurrency and Computation: Practice and Experience (CCPE)*. – 2012. – Vol. 24 (13). – P. 1397-1420.
7. *Mastroianni C., Meo M., Papuzzo G.* Probabilistic Consolidation of Virtual Machines in Self-Organizing Cloud Data Centers // *IEEE Transaction of Cloud Computing*. – 2013. – Vol. 1. – P. 215-228.
8. *Mosa A., Paton N.W.* Optimizing virtual machine placement for energy and SLA in clouds using utility functions // *Journal of Cloud Computing: Advances, Systems and Applications*. – 2016. – Vol. 5.
9. *Monil, Mohammad Alaul Haque and Rahman, Rashedur M.* VM consolidation approach based on heuristics, fuzzy logic, and migration control // *Journal of Cloud Computing: Advances, Systems and Applications*. – 2016. – Vol. 5.
10. *Guenter B., Jain N., and Williams C.* Managing cost, performance, and reliability tradeoffs for energy-aware server provisioning // *Proc. of the 30st Annual IEEE Intl. Conf. on Computer Communications (INFOCOM)*. – 2011. – P. 1332-1340.
11. *Balashov N., Baranov A., Korenkov V.* Optimization of over-provisioned clouds // *Physics of Particles and Nuclei Letters*. – 2016. – Vol. 13, No. 5. – P. 609-612.
12. *McNab A., Stagni F., and Luzzi C.* LHCb experience with running jobs in virtual machines // *J. Phys.: Conf. Ser.* – 2015. – Vol. 664.
13. *Computing Center of the Institute of High Energy Physics (IHEP-CC)* “VCondor – virtual computing resource pool manager based on HTCondor”. – URL: <https://github.com/hepgnu/VCondor> (accessed on: 08.11.2016).
14. *McNab A., Love P., and MacMahon E.* Managing virtual machines with Vac and Vcycle // *J. Phys.: Conf. Ser.* – 2015. – Vol. 664.
15. *Feller E., Rilling L., and Morin C.* Snooze: A scalable and autonomic virtual machine management framework for private Clouds // *Proceedings of the 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*. – 2012. – P. 482-489.
16. *Beloglazov, R. Buyya.* OpenStack Neat: A Framework for Dynamic and Energy-Efficient Consolidation of Virtual Machines in OpenStack Clouds // *Concurrency and Computation: Practice and Experience (CCPE)*. – 2015. – Vol. 27, No. 5. – P. 1310-1333.
17. *Anne-C'ecile Orgerie, Laurent Lef'ever.* When Clouds become Green: the Green Open Cloud Architecture // *International Conference on Parallel Computing (ParCo)*. – 2009. – P. 228-237.
18. *Ward J.S., Barker A.* Observing the clouds: a survey and taxonomy of cloud monitoring // *Journal of Cloud Computing: Advances, Systems and Applications*. – 2014. – Vol. 3.
19. *Ward J.S., Barker A.* Cloud cover: monitoring large-scale clouds with Varanus // *Journal of Cloud Computing: Advances, Systems and Applications*. – 2015. – Vol. 4.
20. *Open Grid Forum* “Open Cloud Computing Interface”. – URL:<http://occi-wg.org/> (accessed on: 23.11.2016).

## REFERENCES

1. *Helge Meinhard.* Virtualization, clouds and IaaS at CERN, *VTDC '12 Proceedings of the 6th international workshop on Virtualization Technologies in Distributed Computing*, ACM New York, NY, USA, 2012, pp. 27-28.
2. *Timm, S. et al.* Cloud Services for the Fermilab Scientific Stakeholders, *J.Phys.Conf.Ser.*, 2015, Vol. 664, No. 2.
3. *Baranov A.V., Balashov N.A., Kutovskiy N.A., Semenov R.N.* JINR cloud infrastructure evolution, *Physics of Particles and Nuclei Letters*, 2016, Vol. 13, No. 5, pp. 672-675.
4. *Feller E., Rilling L., Morin C.* Energy-aware ant colony based workload placement in clouds, *Proceedings of the 12th IEEE/ACM International Conference on Grid Computing*, Lyon, France - 2011.
5. *Farahnakian F., Liljeberg P., Plosila J.* LiRCUP: Linear regression based CPU usage prediction algorithm for live migration of virtual machines in data centers, *39th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA)*, 2013, pp. 357-364.
6. *Beloglazov A., Buyya R.* Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers, *Concurrency and Computation: Practice and Experience (CCPE)*, 2012, Vol. 24 (13), pp. 1397-1420.

7. *Mastroianni C., Meo M., Papuzzo G.* Probabilistic Consolidation of Virtual Machines in Self-Organizing Cloud Data Centers, *IEEE Transaction of Cloud Computing*, 2013, Vol. 1, pp. 215-228.
8. *Mosa A., Paton N.W.* Optimizing virtual machine placement for energy and SLA in clouds using utility functions, *Journal of Cloud Computing: Advances, Systems and Applications*, 2016, Vol. 5.
9. *Monil, Mohammad Alaul Haque and Rahman, Rashedur M.* VM consolidation approach based on heuristics, fuzzy logic, and migration control, *Journal of Cloud Computing: Advances, Systems and Applications*, 2016, Vol. 5.
10. *Guenter B., Jain N., and Williams C.* Managing cost, performance, and reliability tradeoffs for energy-aware server provisioning, *Proc. of the 30th Annual IEEE Intl. Conf. on Computer Communications (INFOCOM)*, 2011, pp. 1332-1340.
11. *Balashov N., Baranov A., Korenkov V.* Optimization of over-provisioned clouds, *Physics of Particles and Nuclei Letters*, 2016, Vol. 13, No. 5, pp. 609-612.
12. *McNab A., Stagni F., and Luzzi C.* LHCb experience with running jobs in virtual machines, *J. Phys.: Conf. Ser.*, 2015, Vol. 664.
13. *Computing Center of the Institute of High Energy Physics (IHEP-CC)* "VCondor – virtual computing resource pool manager based on HTCCondor". Available at: <https://github.com/hepgnu/VCondor> (accessed 08 November 2016).
14. *McNab A., Love P., and MacMahon E.* Managing virtual machines with Vac and Vcycle, *J. Phys.: Conf. Ser.*, 2015, Vol. 664.
15. *Feller E., Rilling L., and Morin C.* Snooze: A scalable and autonomic virtual machine management framework for private Clouds, *Proceedings of the 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*, 2012, pp. 482-489.
16. *Beloglazov, R. Buyya.* OpenStack Neat: A Framework for Dynamic and Energy-Efficient Consolidation of Virtual Machines in OpenStack Clouds, *Concurrency and Computation: Practice and Experience (CCPE)*, 2015, Vol. 27, No. 5, pp. 1310-1333.
17. *Anne-Cécile Orgerie, Laurent Lefever.* When Clouds become Green: the Green Open Cloud Architecture, *International Conference on Parallel Computing (ParCo)*, 2009, pp. 228-237.
18. *Ward J.S., Barker A.* Observing the clouds: a survey and taxonomy of cloud monitoring, *Journal of Cloud Computing: Advances, Systems and Applications*, 2014, Vol. 3.
19. *Ward J.S., Barker A.* Cloud cover: monitoring large-scale clouds with Varanus, *Journal of Cloud Computing: Advances, Systems and Applications*, 2015, Vol. 4.
20. *Open Grid Forum* "Open Cloud Computing Interface". Available at: <http://occi-wg.org/> (accessed 23 November 2016).

Статью рекомендовал к опубликованию д.ф.-м.н. А.И. Аветисян.

**Балашов Никита Александрович** – Объединенный институт ядерных исследований; e-mail: [balashov@jinr.ru](mailto:balashov@jinr.ru); 141980, Московская обл., г. Дубна, ул. Жолио-Кюри, 6; инженер-программист 1 категории.

**Баранов Александр Владимирович** – e-mail: [baranov@jinr.ru](mailto:baranov@jinr.ru); инженер-программист 1 категории.

**Кадочников Иван Сергеевич** – e-mail: [kadivas@jinr.ru](mailto:kadivas@jinr.ru); инженер-программист.

**Кореньков Владимир Васильевич** – e-mail: [korenkov@jinr.ru](mailto:korenkov@jinr.ru); директор лаборатории; д.т.н.; с.н.с.

**Кутовский Николай Александрович** – e-mail: [kut@jinr.ru](mailto:kut@jinr.ru); научный сотрудник; к.ф.-м.н.

**Нечаевский Андрей Васильевич** – e-mail: [nechav@jinr.ru](mailto:nechav@jinr.ru); инженер-программист 1 категории.

**Пелеванюк Игорь Станиславович** – e-mail: [pelevanyuk@jinr.ru](mailto:pelevanyuk@jinr.ru); инженер-программист.

**Balashov Nikita Alexandrovich** – Joint Institute for Nuclear Research; e-mail: [balashov@jinr.ru](mailto:balashov@jinr.ru); 6, Joliot-Curie street, Dubna, Moscow region, 141980, Russia; software engineer.

**Baranov Alexandr Vladimirovich** – e-mail: [baranov@jinr.ru](mailto:baranov@jinr.ru); software engineer.

**Kadochnikov Ivan Sergeevich** – e-mail: kadivas@jinr.ru; software engineer.

**Korenkov Vladimir Vasilevich** – e-mail: korenkov@jinr.ru; director of the laboratory; dr. of eng. sc.; senior researcher.

**Kutovskiy Nikolay Alexandrovich** – e-mail: kut@jinr.ru; researcher; cand. of phis.-math. sc.

**Nechaevskiy Andrey Vasilevich** – e-mail: nechav@jinr.ru; software engineer.

**Pelevanyuk Igor Stanislavovich** – e-mail: pelevanyuk@jinr.ru; software engineer.

УДК 519.687:519.688:519.673:519.63

DOI 10.18522/2311-3103-2016-12-103114

**М.В. Якобовский, А.А. Бондаренко, А.В. Выродов, С.К. Григорьев,  
М.А. Корнилина, А.И. Плотников, С.В. Поляков, И.В. Попов, Д.В. Пузырьков,  
С.А. Суков**

### **ОБЛАЧНЫЙ СЕРВИС ДЛЯ РЕШЕНИЯ МНОГОМАСШТАБНЫХ ЗАДАЧ НАНОТЕХНОЛОГИИ НА КЛАСТЕРАХ И СУПЕРКОМПЬЮТЕРАХ\***

*В настоящее время научно-технический прогресс связан с решением актуальных проблем нанотехнологии. Большая часть задач нанотехнологии исследуется методами математического моделирования с помощью высокопроизводительной вычислительной техники. В последние годы такое моделирование проводится в рамках облачных технологий. Во всем мире создаются платные и бесплатные облачные сервисы, предоставляющие широкие возможности для моделирования свойств микро- и наносистем, имеющих перспективы применения в различных отраслях промышленности. В данной работе представляется облачный сервис, предназначенный для решения на кластерных и суперкомпьютерных системах многомасштабных задач нанотехнологии актуальных для развития современной электроники. Мотивацией к созданию облачного сервиса была необходимость интеграции идей и знаний по данной прикладной проблеме, специалистов по решению задач данного класса на суперкомпьютерных системах, различных технологий моделирования и множества пакетов прикладных программ, а также различных вычислительных ресурсов, имеющихся у авторов и их партнеров. В результате создан прототип облачной среды, реализованный в виде сервиса KIAM Multilogin и прикладного программного обеспечения доступного из виртуальных машин пользователей. Первым приложением сервиса стал комплекс параллельных программ для суперкомпьютерного моделирования нелинейных газодинамических процессов в микроканалах технических систем. На его примере удалось исследовать проблемы и определить пути реализации систематических суперкомпьютерных расчетов задач данного класса. Предложенная реализация сервиса позволила обеспечить отказоустойчивые масштабные вычисления нескольких вариантов задачи на множестве кластеров и суперкомпьютеров, имеющих различное месторасположение и архитектуру, а также провести детальный анализ полученных численных результатов. Разработанные средства позволяют выполнять расчеты с минимальным вмешательством пользователя. Обеспечивается автоматическая миграция данных между вычислительными кластерами и перезапуск программ в целях продолжения расчета. Промежуточные и результирующие данные с активной вычислительной системы передаются в фоновом режиме в централизованное хранилище. Контрольные точки дополнительно копируются на все дополнительные вычислительные системы. Анализ работы среды управления KIAM Job\_Control показал, что потери времени, связанные с передачей данных между кластерами и перезапуском задач, сократились примерно в 2.5–3 раза по сравнению с ручным управлением со стороны пользователя.*

*Облачный сервис; виртуализация; управление ресурсами; задачи нанотехнологии; многомасштабное компьютерное моделирование нелинейных газодинамических процессов в микроканалах технических систем.*

\* Работа выполнена при поддержке Российского фонда фундаментальных исследований (проекты №№ 15-29-07090-офи\_м, 15-07-06082-а, 16-07-00519-а).