

В.И. Гриценко, А.А. Урсатьев, А.П. Лозинский

## Облачные технологии Многоцелевых комплексов геораспределенных систем

Исследованы свойства облачных платформ. Описано архитектурное решение инфраструктуры Многоцелевого комплекса обработки научных данных и исследований как составной части геораспределенных систем с динамическим перераспределением ресурсов, развитыми системами хранения данных и телекоммуникаций, позволяющими оперировать с различными информационными средами.

The cloud computing features are researched. The architectural decision on the infrastructure of the cloud model for IT-services rendering with the dynamic resources rearrangement, the developed systems of data and telecommunication storage enabling to operate with the different information medium is described.

Досліджено властивості платформ, призначених для хмарних обчислювань. Описано архітектурне рішення інфраструктури Багатоцільового комплексу обробки наукових даних і досліджень як складової частини георозподілених систем з динамічним перерозподілом ресурсів, розвиненими системами зберігання даних і телекомунікацій, які дозволяють оперувати з різними інформаційними середовищами.

**Введение.** В последнее десятилетие все чаще реализация различного рода и назначения ИТ-систем происходит с привлечением облачных решений [1, 2]. Это программные решения систем управления бизнесом *CRM* и *ERP* [1]; создание научных организаций нового типа – распределенных в пространстве исследовательских центров [2], виртуальных коллективов, использующих, в том числе, и геораспределенные ресурсы при решении общей задачи [3]; центров обработки научных данных на платформе «инфраструктура как сервис» [3, 4]; компьютерное моделирование научных и инженерных задач в «облаке» – новые формы передачи завершенных научных разработок программных комплексов в виде услуг по их эксплуатации [5]; создание инструментальной среды для анализа данных секвенирования<sup>1</sup> молекул микроРНК

на базе облачных сервисов [6] вплоть до привлечения услуг вычислительного облака для обработки и анализа данных, расположенного за пределами научной организации, вместо того, чтобы бесконечно обновлять ИТ-инфраструктуру [7], и многое другое.

### Возможные решения

Как видим, подходы здесь различны – от совершенных программных сред, предоставляемых потребителю по сервисной модели *SaaS*, до инфраструктуры по модели *IaaS* [8, 9]. В последнем случае потребителю предоставляются средства обработки и хранения данных, сетей и других базовых вычислительных ресурсов, используя которые потребитель может развертывать и выполнять произвольное программное обеспечение (ПО), включая операционные системы и приложения. Он может контролировать операционную систему (ОС), средства хранения, развертываемые приложения и, возможно, владеть ограниченным контролем над выбранными сетевыми компонентами [8].

В тех случаях, когда потребитель вычислительных ресурсов имеет свой парк оборудования, разрозненного, рассредоточенного по подразделениям и требующего периодического обновления, возможно иное решение, продикто-

---

<sup>1</sup> После публикации первого генома человека секвенирование ДНК превратилось в один из повседневных инструментов молекулярной биологии и стало рутинной процедурой. За последние годы появилось значительное число облачных сервисов, разработанных для анализа данных секвенирования. Большая их часть отличается высокой коммерческой стоимостью, бесплатные же аналоги предоставляют недостаточные возможности, а также имеют ограничения на входные объемы анализируемых данных [6].

ванное низкой эффективностью затрат на ПО и дорогостоящую технику вследствие их разобщенности, неравномерной загрузки и неразвитого сервиса. Отсутствие, как правило, централизованного управления локальными компонентами ИТ-инфраструктуры, мониторинга оборудования и ПО, управления обновлениями и других возможностей, как следствие, повышают эксплуатационные расходы. Неизбежное бесконтрольное разрастание инфраструктуры еще более усугубляет положение.

Выходом из создавшегося положения, как это ни парадоксально, может послужить концентрация вычислительных ресурсов и привлечение новых прогрессивных технологий, в частности, – виртуализации в широком смысле этого понятия и, в последующем, облачных решений, обеспечивающих динамическое перераспределение ресурсов [9, 10]. Ресурсы можно представить в виде трех основных компонент: вычислительные средства, система хранения данных (СХД), средства коммуникации и сети. Технологии виртуализации позволяют собирать компоненты в единый пул с последующим гибким перераспределением этих ресурсов с тем, чтобы эффективно их использовать. Так, замена физических машин на виртуальные позволит поднять эффективность использования процессорных ресурсов серверов вследствие одновременного запуска на одной физической машине разных версий или одной ОС, давая возможность серверу выполнять сразу несколько задач. Виртуализация сети, достигаемая архитектурной моделью *SDN* (*software-defined network* – программно-конфигурируемые сети) путем разделения функций управления и переадресации, позволяет организовывать более гибкую и мобильную сетевую инфраструктуру. Традиционные сети сконфигурированы статически, и любое изменение сетевой конфигурации требует ручного вмешательства, в то время как во многих современных задачах сетевая инфраструктура должна оперативно меняться, адекватно реагируя на поступающие запросы [11, 12].

Предлагаемое решение не ново, оно сродни конвергентным инфраструктурам (*Converged Infrastructure, CI*) [13, 14]. Характерная их осо-

бенность в том, что технология *CI* прежде всего позволяет собрать вместе вычислительные мощности разобщенных серверов, СХД, сетей и коммуникационного оборудования в пул ресурсов. В состав ресурсного пула могут быть включены как виртуальные, так и физические ресурсы, например серверы с установленными приложениями, которые нельзя перемещать в виртуальную среду. *CI* – это прежде всего централизация и виртуализация, которые позволяют получить наибольшие преимущества от виртуализации и обеспечить максимальную динамичность. Обычно под конвергентной инфраструктурой понимают общее законченное решение, собранное из взаимодополняющих готовых компонентов и специального ПО для управления инфраструктурой, включающего в себя среду разработки и автоматизации процессов управления виртуальной инфраструктурой – *Orchestrator* [9]. ПО, обеспечивающее централизованное управление обычно виртуализованными и объединенными в общие пулы ресурсами, распределяемыми между различными приложениями, повышает экономическую эффективность и снижает стоимость эксплуатации. Обычно подразумевается три основных стимула к использованию *CI*: моральное и материальное старение инфраструктуры, зрелость технологий виртуализации и признание неоспоримых преимуществ частных облаков [9]. Конвергентные инфраструктуры получили стремительное развитие, и крупные корпорации на рынке ИТ включили *CI* в зону своего внимания.

Динамические инфраструктуры облака позволяют по-новому взглянуть на функционирование геораспределенных систем. Из известных моделей развертывания [8] только локальному частному облаку присуща ограниченность ресурсов, определяемая ожидаемой максимальной нагрузкой. И только в случае, когда поддерживается достаточное разнообразие нагрузок, локальное частное облако, вероятно, в состоянии обеспечить эластичность для клиентов организации в пределах заявленной мощности. Гибридное облако, напротив, образует композицию из двух и более облаков (общественных, публичных или частных), остающихся уникаль-

ми по своей сущности, но объединенными стандартизированными или проприетарными технологиями, обеспечивающими переносимость (*portability*) данных и приложений между облаками (например, такими технологиями, как пакетная передача данных для балансировки нагрузки между облаками) [8]. В пределах этой композиции из облаков – распределенной, в том числе и географически, компьютерной среды, физические ресурсы которой организованы в виде пула с возможностью динамического назначения и переназначения различных физических и виртуальных ресурсов в соответствии с потребностями, можно перераспределять вычислительные мощности. Это, по мнению авторов, именно та ситуация, которая необходима современным геораспределенным системам и виртуальным производствам – «перемещать» рабочие нагрузки между площадками, организовывать виртуальные коллективы, центры научных исследований без приобретения и развертывания дорогостоящих ПО и оборудования с возможностью привлечения информационных ресурсов разных географических локаций. «Перемещение» ресурсов, как собственно и их организация в пул, оперативная эластичность (*rapid elasticity*), т.е. способность быстрого изменения вычислительных возможностей как в сторону увеличения, так и уменьшения, в зависимости от масштабов потребления, выполняется специальным ПО, присущим облачным ОС, например [15, 16]. Это позволяет выделять ресурсы под рабочие нагрузки, не привязывая задачи к конкретному оборудованию и масштабам его ресурсов. Осуществляется это следующим образом.

### Механизм динамического перераспределения ресурсов «облака»

В [8] приведен механизм управления динамическим перераспределением ресурсов облачной инфраструктуры под возникающие нагрузки. Он иллюстрируется многоуровневой абстрактной моделью *IaaS* инфраструктуры облака (рис. 1). *IaaS* облака – это компьютерная система для динамической аренды ресурсов. Пользовательские запросы и команды поступают на верхний уровень системы и перенаправляются вниз на уровни, которые или отвечают на запрос, или выпол-

няют команду. Поток отчетов о состоянии в обратном направлении возвращается к пользователю. В случае необходимости привлечения извне дополнительных ресурсов или расширения облака менеджер облака подключается к скоростному Интернету. Общение между компьютерными менеджерами, как правило, локальное и высокоскоростное.

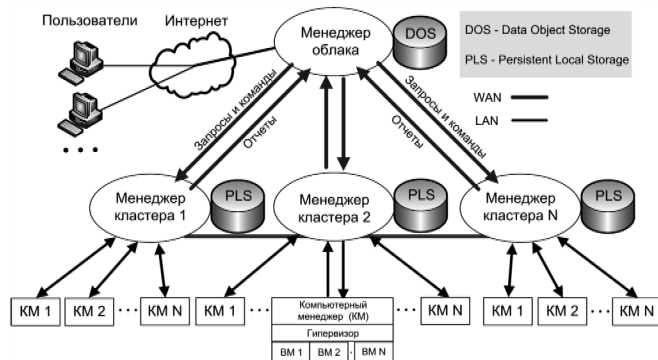


Рис. 1. Логическая облачная архитектура *IaaS*

На верхнем уровне менеджер облака ответствен за учетные записи пользователей и высокоуровневое распределение ресурсов в общем облаке. На среднем уровне – менеджеры ( $МК_j$ ) наряду с локальными блочными хранилищами (*PLS*) ответственны за компьютеры в кластерах и их взаимосвязи. Нижний уровень (см. рис. 1) ответствен за запуск виртуальной машины (*VM*) на индивидуальных компьютерах – менеджеры  $КМ_j$ . Конкретная реализация может разделить и распараллелить некоторые компоненты для повышения производительности, может ввести больше промежуточных слоев для дополнительного согласования или разместить хранилища в сетях, отличных от тех, что указаны в абстрактной модели [8].

Менеджер облака (*МО*) – это точка публичного доступа к облаку и, координируя действия с объектным хранилищем (*DOS*), обеспечивает аутентификацию и авторизацию пользователей. Пользователи в соответствии с учетными записями управляют арендованными ресурсами. С другой стороны, использование сервиса *DOS* позволяет отслеживать правильность идентификации пользователей, чтобы разрешить им выполнять административные действия в хранилище, а также для выставления счета за услуги. Непре-

менное условие для *DOS* – доступность его услуг для ВМ облака и для систем вне облака. Это ограничение предполагает непосредственное размещение *DOS* с менеджером облака и предоставление им доступа к глобальной вычислительной сети.

МО также выполняет размещение ресурсов на верхнем уровне, когда потребитель дает команду арендовать значительное количество ресурсов. При этом он должен определить, имеется ли в облаке достаточное количество ресурсов, чтобы удовлетворить запрос, и при положительном исходе проверить, какой КМ (или менеджеры) имеют ресурсы или их часть. Если запрос может быть удовлетворен, МО должен распределить ресурсы среди участвующих МК<sub>j</sub> и координировать установку виртуальной сети так, чтобы потребитель имел равнозначный (одинаковый) доступ ко всем ресурсам. МО так же будет проводить любые глобальные политики, регулирующие запросы к ресурсам.

Менеджер кластера (МК<sub>j</sub>) отвечает за работу группы компьютеров, соединенных посредством высокоскоростной сети. Компьютерный кластер может содержать десятки и сотни компьютеров. МК<sub>j</sub> получает команды на размещение ресурсов и запросы от МО и определяет, какую часть или весь запрос он может удовлетворить, используя ресурсы компьютеров кластера. МК запрашивает у менеджера компьютера (КМ<sub>j</sub>) доступность ресурсов и возвращает сообщение МО. Если затем МО отдает команду на выделение ресурсов, то МК<sub>j</sub> инструктирует компьютерного менеджера относительно распределения ресурсов и перенастройки виртуальной сетевой инфраструктуры с тем, чтобы дать потребителю равномерный доступ ко всем компьютерам.

Каждый менеджер кластера (см. рис. 1) присоединен к постоянному локальному хранилищу (*PLS*)<sup>2</sup>. Виртуальным машинам необходимо по-

<sup>2</sup> Например, блочное хранение данных *Cinder* в системе *OpenStack* [17]. Блочное хранение в «облаке» это принципиально новый концепт: высокопроизводительный сервис, позволяющий *OpenStack* снабжать пользователя любыми необходимыми данными на постоянной основе. *Cinder* отлично абстрагирует различные типы хранилищ данных. Одно из главных отличий програм-

стоянное дископодобное блочное хранилище для образов ВМ на время их отключения. Наиболее приемлемым местом для *PLS* есть МК<sub>j</sub> со скоростным соединением с ВМ (например, *10 GB Ethernet*). На размещение положительно влияет и тот фактор, что хранилище не зависит от какой-либо компьютерной системы.

На нижнем слое иерархии компьютерный менеджер кооперируется с гипервизором, который запускается на каждой компьютерной системе в кластере. В ответ на запросы менеджера кластера КМ<sub>j</sub> возвращает отчет с информацией о количестве запущенных и свободных ВМ. Используя командный интерфейс гипервизора, КМ<sub>j</sub> в соответствии с исходящими от менеджера кластера командами перенастраивает ВМ (запуск, остановка, приостановка) и настраивает конфигурацию локальной виртуальной сети. Он также отвечает за оптимизацию передачи сетевых пакетов разными ВМ, запущенными на одном гипервизоре, направленную на увеличение производительности [8].

Таким образом, работа инфраструктуры облака *IaaS* (см. рис. 1) – это циклический процесс потока потребительских запросов, проходящих по иерархии, и потока ответов, возвращающихся потребителю. Последние, помимо предоставленных им в пользование ВМ, могут получить непосредственный доступ к серверу хранения данных в облаке. Несмотря на то, что совокупные всплески и падения запросов должны быть менее ярко выраженными, чем аналогичные параметры индивидуальных пользовательских запросов, облако в ряде случаев может использоваться не в полной мере, и тогда миграция пользовательских нагрузок с одной компьютерной системы на другую, с одного кластера на другую, –

---

мно-управляемого хранилища в том, что можно абстрагироваться от различных типов хранилищ и старых *API* и превратить их в единый ресурс [17].

Напомним, при изменении содержимого файла в блочной системе происходит копирование исключительно образа измененного блока. При создании копии файла, дублирующие блоки удаляются, сохраненными остаются только измененные блоки данных. Это позволяет уменьшить размер копии и объем данных. Вторая особенность блочного хранения – дископодобное хранилище с высокой скоростью работы.

это стратегия, которая сможет сконцентрировать потребительскую нагрузку на уже загруженных машинах и выключить другие серверы, чтобы уменьшить расходы на их содержание и провести техническое обслуживание.

Мобильность виртуальных машин – это существенный инструмент при возникновении неизбежных потребностей замены оборудования без масштабных отключений. Кроме того, провайдеры могут использовать виртуализацию для прозрачного добавления новых мощностей в виде дополнительных компьютеров в кластерах или дополнительных кластеров для удовлетворения растущих нагрузок.

### Облачные решения Многоцелевого комплекса обработки научных данных и исследований

В качестве примера облачного решения рассмотрим проект Многоцелевого комплекса обработки научных данных и исследований (МЦ КОНДИ, или Многоцелевой комплекс) Международного научно-учебного центра информационных технологий и систем. МЦ КОНДИ (рис. 2) неоспоримо будет эффективен при разработке ресурсоемких интеллектуальных ИТ, вытекающих из государственных научно-техни-

ческих программ [18]. Это прежде всего решение оптимизационных задач в экономике, структурное распознавание образов, разработка речевых информационных технологий, новых технологий в биологии и медицине, обработка сигналов сложной формы, управление в робототехнике [19, 20] и многое другое.

Интеллектуальные ИТ – это высокие, наукоемкие ИТ. Основное свойство интеллекта – способность к самообучению, таким образом, интеллектуальным ИТ должно быть присуще изменение модели поведения при воздействии различных наборов внешних возмущений. Эти и другие функциональные особенности интеллектуальных ИТ позволяют достаточно быстро создавать и развивать распределенные информационные ресурсы различного уровня и назначения, чем достигается выбор экономических инфраструктур и моделей информатизации в рамках общих программ создания и развития информационного общества [9].

Современные исследования в значительной степени базируются на математическом (компьютерном) моделировании, которое для многих академических учреждений и исследовательских организаций служит основным инструментом,



Рис. 2. Функционально-логическая схема МЦ КОНДИ

базовой вычислительной инструментальной средой проводимых исследований [21, 22]. В этом контексте МЦ КОНДИ представлен рядом моделирующих комплексов, направленных на исследование и разработку процессов обучения, электронного управления, информационно-коммуникационных технологий (ИКТ) и др. В обучении следует выделить альтернативные электронные модели образования, соответствующие требованиям всеобщего доступа к обучению, непрерывности и трансграничности [23]. В рамках задач электронного управления – проекты электронного правительства, электронного взаимодействия структур различных форм собственности, в которых особую роль приобретают модели, обеспечивающие прозрачность и коллективное электронное управление [9]. Интеграция разнородных систем в гетерогенной среде широкополосных сетей, интеллектуальное управление в современных высокоскоростных сетях следующего поколения выдвигают первоочередные задачи по созданию знаниеориентированных интеллектуальных ИКТ: разработка моделей, алгоритмов и методов семантической интероперабельности; семантический поиск, распознавание и интерпретация объектов информационного пространства на основе онтологического представления знаний о предметной области [24].

Кроме того, моделирующие комплексы позволяют выполнить такие необходимые для народного хозяйства страны разработки, как создание и/или исследование моделей тех или иных природных явлений, например, в гидрометеорологии, гидрографии и др. [25, 26], технологических систем различной конфигурации и степени сложности, разработку и отладку специального программного обеспечения, работы по оптимизации решения ряда практических задач и пр.

Исследования, моделирование большинства задач, дистанционное обучение, функционирование виртуальных коллективов сегодня немыслимы без взаимодействия с информационными средами (см. рис. 2), с многократно возросшими объемами разнородных данных и информационной насыщенностью, их тесной взаимосвязью с рассматриваемой проблематикой. Этому в не малой степени способствовало массовое появление

высокоскоростных каналов связи и развитие высокопроизводительных коммуникаций. Примером тому может послужить доступность в Украине сетей *GEANT*<sup>3</sup> и *Grid* [27–29]. Известно также создание национальной сети *ChANT* (*Chernogolovka Academic Network*) для информационно-технологического обеспечения фундаментальных научных исследований [30] со статусом *LIR*<sup>4</sup>. Можно сказать, сети создали инфраструктуру для новых развивающихся информационных технологий и процессов обучения.

Исследования также должны опираться на электронные ресурсы информационных фондов Национальной библиотеки (см. рис. 2) и библиотек учебных заведений Украины, видеоматериалов, Фонда технологий и интеллектуальной собственности и др.

Непременным атрибутом процессов обучения, в том числе и дистанционного, функционирования распределенных виртуальных коллективов, проведения семинаров, международных научных конференций и другого должны быть голосовая *VoIP* и видеоконференцсвязь, так как непосредственное общение выполняет немаловажную роль. Реализация последней сопряжена с определенными трудностями, вызванными противоречиями между конкретными требованиями к разрешению видео, определяющему плотность видеопотока, ограниченной пропускной способностью каналов связи и производительностью систем обработки мультимедиа трафика. Это противоречие усугубляется с ростом числа пользователей. При выборе платформы разработки видеоконференцсвязи или готовых к использованию аппаратных систем необходима уверенность, что будет соблюдена интероперабельность в распределенной среде партнеров и возможность реализовать поставленные задачи.

---

<sup>3</sup> *GEANT* – Европейская мультигигабитная научно-образовательная сеть, объединяющая более 8000 научных и более 40 млн. пользователей. Сеть *GEANT* позволяет всем ее участникам проводить совместные научные исследования, кооперироваться для научной деятельности и внедрять образовательные программы.

<sup>4</sup> Технически это означает самостоятельную маршрутизацию потоков информации сети *ChANT* с другими сетями, составляющими глобальную сеть Интернет [30].

Предполагается использование системы на разных масштабах: индивидуальное общение; видеолекция с использованием многих экранов или большого экрана с высоким разрешением; отображение результатов моделирования на экране формата 4K; конференция международного масштаба с передачей информации от всех участников к каждому и пр.

Корпоративная (кампусная либо локальная) сеть организации (см. рис. 2) обеспечивает доступ пользователей к ресурсам комплекса и к внешнему миру, включая международные информационные среды, в том числе, упомянутые *GEANT*, *Grid*, *UNESCO*. Центр управления сетью *NOC (Network Operational Center)*<sup>5</sup> кроме стандартных задач мониторинга и разграничения доступа, наделен также функцией коммутации системы взаимодействия и отображения информации (см. рис. 2) в зависимости от задач, выполняемых комплексом, и определенной интеллектуализации взаимодействия пользователей и подсистем Многоцелевого комплекса. Например, пользователь может выполнять определенные операции на моделирующем комплексе, в процессе которых он, пользуясь справочно-технологической БД *NOC*, формирует адресные запросы к информационным фондам и использует полученную информацию для выполнения операций, заданных пользователем (рис. 3). Справочно-технологическая БД поддерживается в актуальном состоянии посредством служебных запросов к информационным фондам и средам.

Состав функциональных компонент Многоцелевого комплекса (см. рис. 2), организованных в виде пула вычислительных ресурсов – ПВР (мощности серверов, системы хранения, сетевое оборудование и др.), СУР, обеспечивающая распределение этих ресурсов по за-

просу пользователей, сложное и разнообразное информационное окружение, с которым посредством *NOC* взаимодействуют потребитель и подсистемы комплекса, все это обуславливает требование достаточно глубокой внутренней интеграция *NOC*, СУР и ПВР. Этим обеспечивается минимизация задержек в получении/обработке информации, актуализации БД и, соответственно, снижение нагрузки на сеть.

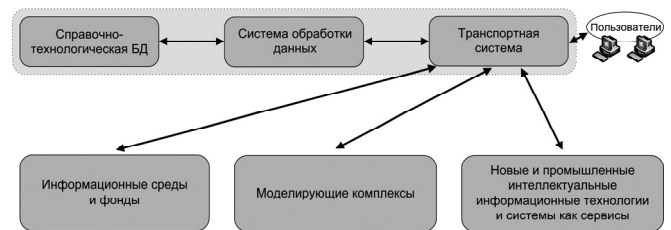


Рис. 3. Схема взаимодействия *NOC* с компонентами комплекса

В этой связи распространенное решение к проектированию *NOC* с использованием выделенного аппаратного обеспечения предложено заменить виртуальными машинами из ПВР, т.е. позиционировать его в составе Многоцелевого комплекса (рис. 4). Такой подход также повышает надежность комплекса в целом при сохранении допустимого уровня информационной безопасности.

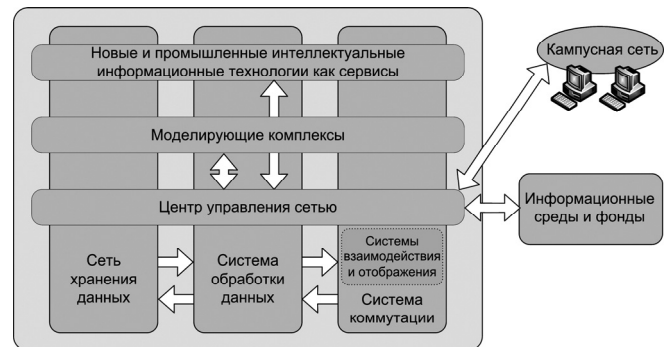


Рис. 4. Состав и схема взаимодействия компонент МЦ КОНДИ

В рамках приведенной функциональности комплекса особые требования предъявляются к хранению информации. Хранению подвергаются разнородные данные – исходные для исследований, данные промежуточных и конечных результатов, программный инструментарий исследований и пр. При этом следует учитывать, что обработка значительных массивов данных требует повышенного быстродействия системы хранения, а при длительном их хранении общий объ-

<sup>5</sup> Центр управления сетью *NOC* – в традиционном понимании, здание либо помещение, в котором расположены управляющие системы и из которого проводится комплекс мероприятий по управлению и мониторингу телекоммуникационной либо компьютерной сети [31]. Иногда этим термином называется собственно техническое решение, предназначенное для указанных целей управления/мониторинга, например [32].

ем (исходных данных, окончательных результатов, справочных и архивных материалов, мультимедийных материалов, используемых в обучении и др.) обуславливает необходимость использования накопителей повышенного объема.

Распределение вычислительных мощностей из ПВР в соответствии с заявленными задачами осуществляется СУР (см. рис. 2), реализуемой специальным ПО облачной ОС, например [15]. Механизм динамического перераспределения ресурсов пула под возникающие нагрузки аналогичен управлению инфраструктурой облака (см. рис. 1).

**Предполагаемая реализация.** Оборудование МЦ КОНДИ можно представить в виде трех основных систем (см. рис. 4): хранения, обработки данных и транспортно-коммуникационной, а также некоторых вспомогательных подсистем (например, конференцсвязи) и инженерной инфраструктуры обеспечения (табл. 1).

**Таблица 1.** Основные параметры оборудования МЦ КОНДИ

<b>1</b>	<b>Вычислительная система (64 ядра 2,2 ГГц, 1 Тб оперативной памяти)</b>
	Серверы-лезвия 10 <i>Gb FLB CTO Blade</i> 4 процессора <i>Intel Xeon E5-4620</i> (8 ядер, 2,2 ГГц 16 Мб кэш); 512 Гб оперативной памяти; 2 × 240 Гб жестких дисков; 24 × 10 Гб оптических сетевых интерфейса  Шасси (8 слотов расширения) 6 × 2400 Вт источников питания; 2 бортовых административных модуля с возможностью горячей замены
<b>2</b>	<b>Система хранения данных (74,4 Тб; 14,4 Тб + 60 Тб)</b>
	Шасси; Полка дисковых накопителей ускоренного доступа 16 × 450 <i>Gb 6 G SAS 10 K</i> ; Полка дисковых накопителей стандартного доступа 36 × 3 Тб 6 <i>G SAS 7,2 K</i> ; Оптические интерфейсы связи с вычислительной системой 8 Гб <i>FC SFP</i>
<b>3</b>	<b>Транспортно-коммуникационная система</b>
	Коммутаторы ядра 2 × 40 Гб; Пограничный маршрутизатор 2 × 100 Мб; Коммутаторы агрегации 48 × 1 Гб; Коммутаторы подключения к вычислительной системе 48 × 1 Гб

*СХД* представляет собой дисковый массив, подключенный к вычислительной системе по четырем оптоволоконным интерфейсам *Fiber Channel* производительностью 8 Гб/с каждый. Объем массива может наращиваться, в зависимости от соотношения количества быстрых и стандартных накопителей и конфигурирования массива, до 864 Тб. Возможно также подключение ленточных систем резервного копирования. Емкость массива разделена на подсистему с опера-

тивным доступом (14,4 Тб) и подсистему стандартного хранения (72 Тб). Система имеет два контроллера накопителей, а также резервированные источники питания. Дисковые накопители соединены через коммутаторы по оптоволоконным каналам с серверами системы обработки данных по схеме «каждый с каждым» (рис. 5), образуя сеть хранения данных (*SAN*) [33, 34].

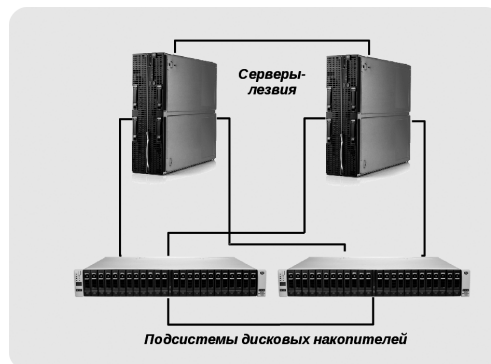


Рис. 5. Сеть хранения данных МЦ КОНДИ

*СХД* с оперативным доступом, кроме использования более быстродействующих накопителей, предполагает конфигурирование массива с чередованием и без резервирования (*RAID*-массив уровня 0); подсистема стандартного хранения в свою очередь содержит накопители большего объема, объединенные в *RAID*-массив с резервированием по четности уровня пять, за счет чего эффективный объем дискового пространства уменьшается до 60 Гб [35].

**Система обработки данных.** Базовые элементы – серверы-лезвия, устанавливаемые в шасси. Предполагается установка двух четырехпроцессорных серверов с оперативной памятью 256 Гб каждый. Допускается расширение до восьми серверов с возможностью увеличения ОП каждого из них до 1 Тб. Тип процессоров и дисков указан в табл. 1. На серверах установлены оптические сетевые интерфейсы для связи с *СХД* и, аналогично, – с коммутационной системой. Серверное шасси содержит два резервированных административных модуля управления и шесть резервированных источников питания. Допускается расширение до восьми таких серверов.

**Транспортно-коммуникационная система** (ТКС) обеспечивает связь системы обработки



данных с рабочими местами пользователей (образуя таким образом корпоративную сеть) с одной стороны – и внешним миром (доступ в Интернет и иные сети) – с другой. В качестве ядра ТКС используются, в зависимости от топологии сети, один или несколько базовых коммутаторов, связывающих между собой разнесенные площадки по оптоволоконному каналу. Коммутаторы содержат резервированные модули администрирования и источники питания. К установленному в центре обработки данных коммутатору подключены подсеть системы обработки данных, маршрутизатор связи с внешним миром (два порта по 100 Мб, обеспечивающих резервированную связь с Интернет-провайдерами) и коммутатор агрегации рабочих мест на 48 портов на 1 Гб. Аналогичные коммутаторы агрегации подключены к каждому базовому коммутатору, чем обеспечивается интеграция корпоративной сети и доступ к ресурсам Многоцелевого комплекса со всех рабочих мест (рис. 6).

*Инженерной инфраструктурой* предусматриваются источники бесперебойного питания (до 12 кВт), система кондиционирования и пожарная сигнализация.

**Системное программное обеспечение.** Для достижения необходимой устойчивости, надежности и безопасности в качестве системного ПО

предлагается использовать *UNIX*-платформу, а для снижения общей стоимости решения и поддержания лицензионной чистоты – свободно распространяемая ОС *Ubuntu* семейства *Debian LINUX*. Для придания решению необходимой гибкости и универсальности в качестве среды управления и совместного использования ресурсов МЦ КОНДИ предложено применить открытую, расширяемую облачную платформу, основанную на технологиях *OpenStack* (рис. 7).

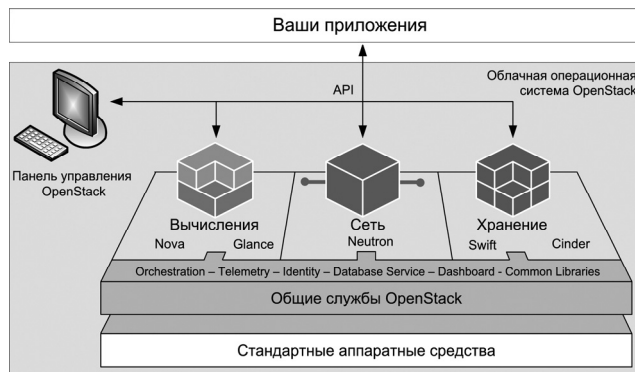


Рис. 7. Архитектура *OpenStack*

Облачная инфраструктура реализуется путем взаимодействия входящих в состав *OpenStack* служб [36]. Каждой службе присвоен одноименный код проекта, в рамках которого она создавалась. Дано также краткое описание служб (табл. 2). В зависимости от решаемой

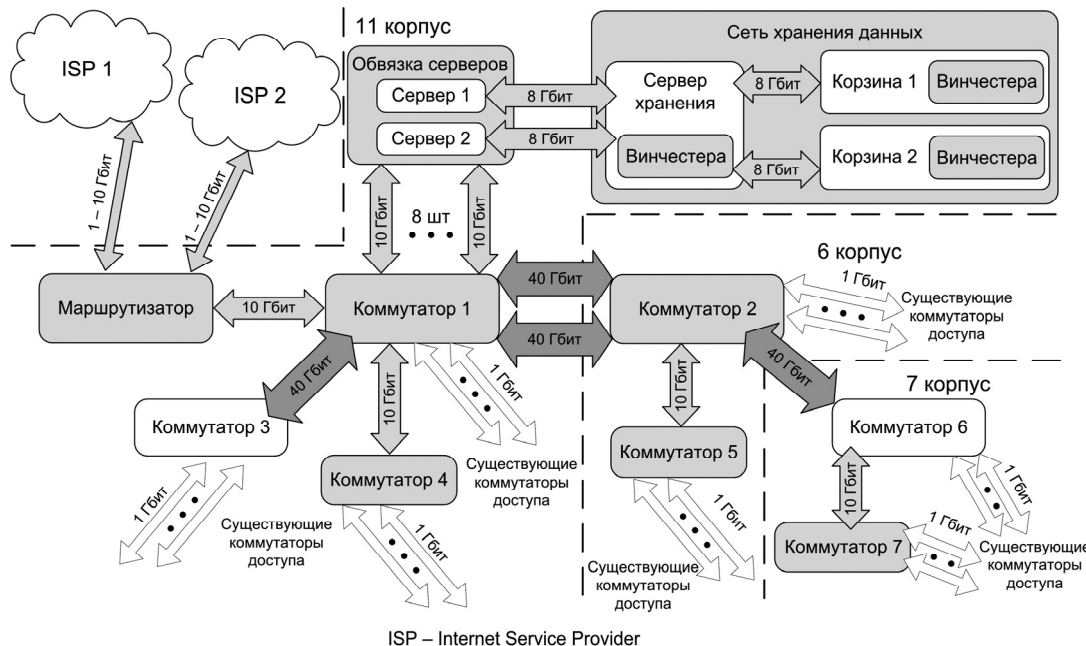


Рис. 6. Реализация Многоцелевого комплекса

задачи возможна установка и использование всех или некоторого подмножества служб. Конечным пользователям для взаимодействия с платформой предоставляется: графический *web*-интерфейс самообслуживания, интерфейс командной строки (*CLI*) и *API* [36].

Таблица 2. Службы *OpenStack Icehouse*

Служба	Описание служб
Интерфейс самообслуживания <i>Horizon</i>	Обеспечивает <i>Web</i> -портал для взаимодействия пользователей и администраторов со службами <i>OpenStack</i> . Графический интерфейс позволяет выполнять запуск <i>VM</i> , назначать им <i>IP</i> -адреса, предоставлять контроль доступа пользователям над выполняемыми проектами, формировать правила для сетевых экранов и др.
Вычисления <i>Nova</i>	Управляет жизненным циклом экземпляров <i>VM</i> в среде <i>OpenStack</i> . В его обязанности входят создание, планирование ресурсов по требованию, запуск и отключение <i>VM</i> . Позволяет управлять рядом гипервизоров [36]: <i>Baremetal</i> , <i>Docker</i> , <i>Hyper-V</i> , <i>Kernel-based Virtual Machine (KVM)</i> , <i>Linux Containers (LXC)</i> , <i>Quick Emulator (QEMU)</i> , <i>User Mode Linux (UML)</i> , <i>VMware vSphere</i> , <i>Xen</i> .
Сеть <i>Neutron</i>	Предоставляет сетевое взаимодействие как услугу службам <i>OpenStack</i> , такой как <i>Nova</i> . Имеет модульную архитектуру, поддерживающую множество популярных сетевых поставщиков и технологий.
Хранение	
Объектное хранилище ( <i>OX</i> ) <i>Swift</i>	<i>OX Swift</i> предоставляет отказоустойчивое масштабируемое распределенное хранилище с резервированием, использующее кластеры серверов [37]. Хранит и предоставляет произвольные неструктурированные объекты данных, используя <i>API</i> на базе <i>RESTful</i> , <i>HTTP</i> .
Блочное хранилище ( <i>BX</i> ) <i>Cinder</i>	Программно-управляемое, дископодобное хранилище с высокой скоростью работы, абстрагирует различные типы хранилищ данных и реализует <i>BX</i> как высокопроизводительную службу [17]. Обеспечивает создание и управление устройствами <i>BX</i> и предоставляет их для постоянного хранения образов запущенных вычислительных узлов <sup>6</sup> .
Общие службы	
Служба идентификации <i>Keystone</i>	Обеспечивает сервис аутентификации и авторизации для других служб <i>OpenStack</i> . Предоставляет каталог доступных сервисов с их конечными точками <i>API</i> – адрес, доступный по сети, обычно описан <i>URL</i> , с которого получают доступ к сервису [38].
Служба хранения образов <i>Glance</i>	Сохраняет и предоставляет образы <i>VM</i> для <i>OpenStack Nova</i> . <i>Nova</i> использует их при инициализации экземпляров <i>VM</i> .
Телеметрия <i>Ceilometer</i>	Выполняет мониторинг и учет использования ресурсов облака для выставления счетов потребителям и маркетинговых исследований: оценки производительности, определения общей нагрузки, масштабируемости облака, накопления статистики и др.

<sup>6</sup> Вычислительный узел представляет собой физический сервер или образ *VM* с необходимыми атрибутами (сетевой адрес, название и др.), готовый к запуску.

Продолжение табл.

Службы более высокого уровня	
Оркестратор <i>Heat</i>	Выполняет задачу сочетания множества составляющих облако приложений путем использования родного формата <i>Heat Orchestration Template (HOT)</i> шаблона [39], либо шаблона <i>AWS CloudFormation</i> посредством <i>API: OpenStack-native REST</i> или <i>CloudFormation-compatible Query</i> .
Облачная СУБД <i>Trove</i>	Обеспечивает масштабируемую и надежную СУБД как услугу, функциональную для реляционных и нереляционных БД.

Функциональные службы *OpenStack* (рис. 8) взаимодействуют посредством сообщений. Обмен сообщениями допускает использование одного из трех сервисов: *RabbitMQ*, *Qpid*, *ZeroMQ* [36, 38]. Для хранения информации службы используют БД СУБД *MySQL*.

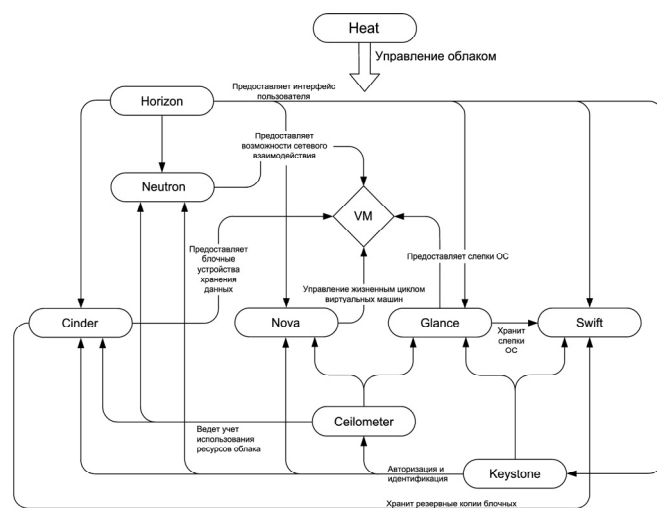


Рис. 8. Концептуальная архитектура *OpenStack*

В комплексах такого рода как МЦ КОНДИ требуется высокая доступность (*High Availability*) предоставляемых услуг. Достигается это созданием кластера, т.е. разновидности параллельной или распределенной системы, состоящей из связанных между собой компьютеров и используемых как единый ресурс. В *OpenStack* построение кластера с холодным резервом (активный/пассивный узлы; активный узел выполняет запросы, а пассивный ждет его отказа и тогда включается в работу) осуществляется путем установки на его узлы программного обеспечения *Pacemaker* [40] – стандарта реализации стеков балансировки нагрузки и высокой доступности на базе платформ *Linux*. Транспортную функцию в кластере выполняет программное обеспечение *Corosync*, обеспечивая надежность обмена служебными сообщениями внутри кластера.

После запуска кластера реализуется высокая доступность служб *OpenStack* путем их дублирования на разных узлах кластера. Для отслеживания служб проводится запуск агентов ресурсов, входящих в состав *Pacemaker*. При выходе из строя узла с активной службой *Pacemaker* по истечении временного интервала таймера активности ресурса запускает резервную службу.

Обеспечение отказоустойчивости *OpenStack Icehouse* реализуется высокой доступностью работы следующих служб [40]: *mysql*, *keystone-api*, *nova-api*, *nova-conductor*, *glance-api*, *neutron-api* и *nova-scheduler*. Эти службы не ориентированы на соединение, т.е. сетевое соединение с ними поддерживается только на время предоставления ответа на поступивший запрос. После предоставления службой ответа сетевое соединение разрывается.

Реализация систем высокой доступности преследует цель уменьшения влияния двух факторов:

- времени простоя системы – возникает в случае недоступности системы для пользователя на протяжении максимально допустимого периода времени;
- потери данных – прецедент удаления или разрушения данных.

Отметим, что подобные решения вычислительных систем, основанные на технологиях *OpenStack*, известны – это *HP CloudSystem* [41]. Они не впервые применяются в научной организации, например, в центре обработки данных Института космических исследований РАН [4], а Европейская организация ядерных исследований *CERN* [42] использует сразу несколько подобных решений [43].

Для ряда задач МЦ КОНДИ есть необходимость использования и других ОС, поддерживаемых оборудованием *FreeBSD UNIX*, ОС семейств *LINUX*, отличных от *Debian*, систем семейства *Microsoft Windows*, проприетарных *UNIX*-платформ (*HP-UX*), etc. Возможность работы пользователей в указанных операционных средах обеспечивается виртуализацией; в качестве гипервизора используется *KVM (LINUX Kernel-based Virtual Machine)* [44].

Применение средств виртуализации позволяет максимально эффективно использовать информационные ресурсы. Особенно это существенно для организаций, как правило, не располагающих таковыми в избытке. Виртуализация обеспечивает:

- гибкое «дробление» ресурсов между пользователями и задачами;
- использование нескольких различных операционных сред, в том числе одним пользователем, с одного рабочего места;
- сравнительно несложное объединение разнородных операционных сред в один виртуальный комплекс;
- концентрацию ресурсов на приоритетных задачах.

### **Работа пользователей на Многоцелевом комплексе**

В процессе работы на многоцелевом комплексе можно применять не совсем еще привычные модели обслуживания. Это, в частности, создание единичных или связанных между собой сетью ВМ, использующих стандартные или кастомизированные ОС, прочие системные программы, для которых создается и особое прикладное ПО. В данном контексте комплекс – это мощная платформа, обеспечивающая работу значительного количества ВМ и виртуальных сетей – собственно и образующих моделирующие комплексы. Одно из самых существенных преимуществ подобной организации вычислительных ресурсов – возможность их распределения между большим количеством задач и пользователей, гибкого управления их изменениями. Как было отмечено, «дробление» ресурсов с целью увеличения возможного количества пользователей и концентрация необходимых мощностей на приоритетных задачах, что практически невыполнимо в пределах традиционных наборов ПК различной мощности, возраста и технического состояния.

Работа пользователей организована непосредственно на серверах комплекса. При этом они будут использовать тонкий клиент [9] либо свой локальный ПК в качестве средства доступа к терминальным серверам, либо отдельным ВМ с предустановленными ОС и необходимым

набором приложений. Здесь подразумевается различное прикладное ПО проектирования – в частности, средства разработки и отладки программ (в том числе и в среде *OpenStack* [45]), готовое прикладное ПО собственной разработки, а также традиционные программы для офисной работы: редакторы, электронные таблицы, системы компьютерного проектирования и пр. Приложения будут запускаться из сред ОС терминальных серверов или виртуальных машин. Конечные пользователи получают доступ к своим приложениям и данным с различных клиентских устройств (вплоть до мобильных). При этом предполагается конфигурирование клиентской среды сообразно удобству работы конкретного пользователя. Так же осуществляется перенос рабочих мест сотрудников офисных и других подразделений в виртуальную среду, или иначе, на виртуальные десктопы.

Пользователь, со своего рабочего места, при старте попадает на внутренний портал, где предусмотрены возможности конфигурирования рабочего места: входа в информационные среды, работы в тех или иных приложениях (с дальнейшим подключением при начале работы сразу в заранее сконфигурированную среду). Комплекс, в частности, допускает одновременное использование сотрудником нескольких «виртуальных» рабочих мест. Точно так же в свою рабочую *Windows*-среду (допустим, «1С-предприятие») войдет напрямую либо через портал сотрудник планового отдела или бухгалтерии. Аналогично, администратор корпоративной сети, использующий *UNIX*-средства администрирования, сможет параллельно заниматься прорисовкой поэтажных схем в *Windows*-системе проектирования *ArchiCAD* или, скажем, *Microsoft Visio* (рис. 9). Отдельным преимуществом такого решения будет повышенный уровень безопасности, в особенности от подавляющего большинства компьютерных вирусов. Виртуальные *Windows*-среды, при соответствующей организации работы, значительно лучше защищены от вирусных атак, чем традиционные сетевые ПК, особенно в отсутствие качественного (и, соответственно, достаточно дорогого) антивирусного ПО.

Практически, получена инфраструктура комплекса с инструментарием, позволяющим конфигурировать его ресурсы с учетом заявок потребителей. Инфраструктура располагается, как и прежде, внутри собственного периметра безопасности, перенос приложений с персональных ПК не снижает защищенности данных.

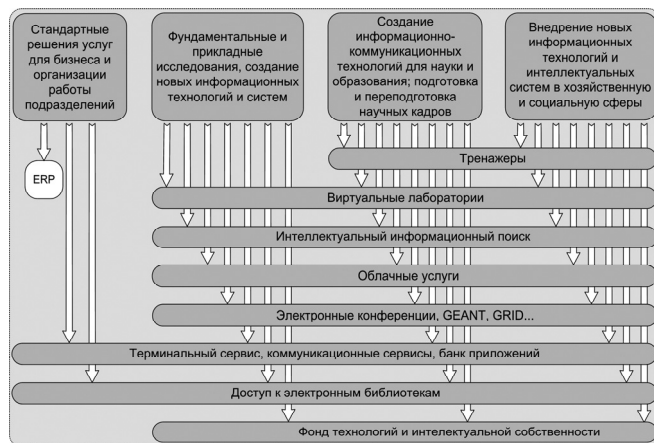


Рис. 9. Архитектура услуг многоцелевого комплекса

Выгоды – централизация вычислительной среды при общем уменьшении численности оборудования и администрирования – высокая надежность и отказоустойчивость. Экономия, достигаемая путем эффективного использования разделяемого пула вычислительных ресурсов, позволяет иметь меньше аппаратных ресурсов<sup>7</sup> при более высокой надежности и отказоустойчивости такой системы. Быстрое восстановление операционных сред – создание копий ВМ и их восстановление из резервных занимает значительно меньше времени. Резервная копия виртуального сервера может быть сразу запущена на другом физическом сервере, чем обеспечивается высокая надежность и доступность ресурса в целом.

Оптимизация достигается и вследствие централизованного администрирования – качество

<sup>7</sup> По статистике большинство серверов при выполнении ими повседневных задач загружены на 15–20 процентов. Использование нескольких виртуальных серверов на одном физическом позволяет увеличить его загруженность до 80 процентов, тем самым, увеличив коэффициент его использования и обеспечив при этом существенную экономию на приобретении аппаратного обеспечения [46].

управления повышается, число администраторов баз данных, сетевых и системных администраторов сокращается. Экономия получается в первую очередь благодаря снижению стоимости эксплуатации ИТ-инфраструктуры и более полному удовлетворению потребностей подразделений организации в сервисах. Происходит переформатирование ИТ-специалистов подразделений во внутреннее сервисное подразделение организации.

**Заключение.** Принятые аппаратные и инфраструктурные решения Многоцелевого комплекса допускают простое наращивание вычислительных ресурсов, а использование открытого ПО проекта *OpenStack*, по мере его освоения, позволит эволюционным путем совершенствовать инфраструктуру<sup>8</sup> и достичь в предоставляемых услугах полноты соответствия модели облачных вычислений [9, 47]: самообслуживание по требованию, свободный сетевой доступ, оперативная эластичность и др. Что касается моделей развертывания (*deployment models*) [8], то ограничимся на первых порах частным облаком (*private cloud*), обеспечивающим исключительное использование единственной организацией, состоящей из разнообразных пользователей, и, как уже упоминалось, подвержено угрозам безопасности не более, чем прежде – до создания комплекса.

Авторы выражают благодарность за вклад в отдельные специальные области настоящей разработки следующим сотрудникам Центра: с.н.с. В. Курочкину, А. Гладуну, И. Трихлебу, м.н.с. Я. Антоноуку и В. Точилину, инж. А. Шевченко.

1. Гриценко В.И., Урсатьев О.А. Інформаційні технології і модель послуг // Науково-технічна інформація. – 2012. – № 3. – С. 47–54.

<sup>8</sup> Напомним, что под облачной инфраструктурой понимают аппаратное и программное обеспечение, которое соответствует характеристикам модели облачных вычислений. Облачную инфраструктуру рассматривают как двухуровневую: физический и уровень абстракции. На первом уровне – аппаратные ресурсы, необходимые для поддержки предоставляемых облачных услуг, – серверы, системы хранения данных и сетевые компоненты. Уровень абстракции состоит из программного обеспечения, развернутого на физическом уровне. Концептуально уровень абстракции находится выше физического уровня.

2. Щур Л.Н., Меньшутин А.Ю., Шикота С.К. Инфокоммуникационное обеспечение исследовательского центра: задачи и инфраструктура // Информационное общество. – 2011. – № 6. – С. 58–65.
3. Шикота С. К задаче развития инфокоммуникационной инфраструктуры научного центра // Там же – 2013. – № 1–2. – С. 102–111.
4. Коноплев В., Назиров Р. «Вычислительное облако» как средство эффективной организации вычислительных ресурсов в центре обработки научных данных // Там же. – С. 17–25.
5. Тарнавский Г.А. Облачные технологии в компьютерном моделировании научных и инженерных задач // Программные продукты и системы. – 2011. – № 2. – <http://www.swsys.ru/index.php?page=article&id=2758>.
6. Использование облачной инфраструктуры для анализа данных секвенирования микроРНК / А. Куриченко, И. Заигрин, Ф. Шарко и др. // Информационное общество – 2013. – № 1–2. – С. 26–38.
7. Геннон Деннис, Рид Дэниэл, Барга Роджер. Облака: демократизация научных вычислений. – <http://www.osp.ru/os/2011/02/13007709/>
8. Cloud Computing Synopsis and Recommendations. Recommendations of the National Institute of Standards and Technology / Badger Lee, Grance Tim, Patt-Corner Robert et al. // NIST Special Publication 800-146. – <http://csrc.nist.gov/publications/nistpubs/800-146/sp800-146.pdf>, May 2012.
9. Гриценко В.И., Урсатьев А.А. Cloud Computing и облачная модель предоставления ИТ-услуг // КВТ. – 2013. – 171. – С. 5–19.
10. Ренессанс виртуализации // Открытые системы. – 2007. – № 2. – С. 1–80.
11. Программно конфигурированные сети // Там же. – 2012. – № 9. – С. 1–64.
12. Программно конфигурированные сети: революция здесь и сейчас // Сети и телекоммуникации. – 2013. – № 3. – С. 38–43.
13. Черняк Л. Время конвергентных инфраструктур. // Открытые системы. – 2012. – № 4. – <http://www.osp.ru/os/2012/04/13015754/>
14. Конвергентные инфраструктуры – корпоративная ИТ-среда настоящего и будущего. – [http://www.karma-group.ru/converged\\_infrastructure\\_article](http://www.karma-group.ru/converged_infrastructure_article), 2014.
15. *OpenStack*: The Open Source Cloud Operating System. – <http://www.openstack.org/software/>
16. *Apache CloudStack* Open Source Cloud Computing. – <http://cloudstack.apache.org/>
17. Блог компании Mirantis/OpenStack. Интервью с Джоном Гриффитом, руководителем проекта OpenStack Cinder (блочное хранение данных). – [http://habrahabr.ru/company/mirantis\\_openstack/blog/186400/](http://habrahabr.ru/company/mirantis_openstack/blog/186400/), 11 июля 2013.
18. Гриценко В.И., Урсатьев А.А. Информационные технологии: тенденция, пути развития // УСиМ. – 2011. – № 5. – С. 3–20.

19. Сухоручкина О.Н. Структуры функциональной организации интеллектуализированного управления мобильной системой // УСиМ. – 2007. – № 3. – С. 26–33, 63.
20. Сухоручкина О.Н. Активирующая подсистема интеллектуального управления сервисным роботом / Экстремальная робототехника, 25–26 сент. 2012 г., С.-Петербург // Сб. докл. Всерос. науч.-техн. конф. – С.-Петербург: Политехника-сервис, 2012. – С. 101–105.
21. Самарский А.А., Михайлов А.П. Математическое моделирование: Идеи. Методы. – М.: Физматлит, 2001. – 320 с.
22. Ильин В. Экзафлопсы против математического моделирования. // Открытые системы. – 2013. – № 5. – С. 16–19.
23. Гриценко В.И. Фундаментальные проблемы E-обучения. – К.: Академперіодика, 2008. – 38 с.
24. Гриценко В.И., Гладун А.Я., Рогушина Ю.В. Семантическое распознавание информационных объектов на основе онтологического представления знаний о предметной области в задачах интеллектуального управления // КВТ. – 2014. – 178. – С. 5–20.
25. Гриценко В.И., Урсатьев А.А. Распределенные информационные системы широкого применения. Концепция. Опыт разработки и внедрения. – К.: Наук. думка, 2005. – 317 с.
26. Using of information resources of Hydrometeorological service of Ukraine for purposes of Water resources assessment and Hydrological forecasting / Ye. Kotikov, V. Gritsenko, S. Krivenko et al. // On the Hydrological Forecasting and Hydrological Bases of Water Management, Bucharest, Romania, 2–6 Sept. 2002. Abstracts 21st Conf. of the Danubian Countries. National Inst. of Meteor. and Hydr., Bucharest, Romania. – 2002. – P. 126; 166 p.
27. GEANT. – <http://www.uran.net.ua/projects/geant/first.htm>, 13.09.2012.
28. Загородний А. Український національний ґрид – проблеми і перспективи – «Український Національний Ґрид – 2012» Робоча нарада, 1–2 лист. 2012 р. – [http://ung.in.ua/upload/user\\_files/UNG\\_Workshop//november\\_1/Zagorodniy\\_UNG.pptx](http://ung.in.ua/upload/user_files/UNG_Workshop//november_1/Zagorodniy_UNG.pptx)
29. UA-Grid: Украинская национальная грид-программа / А.Г. Загородний, С.Я. Свистунов, Л.Ф. Белоус и др. // Матеріали міжнар. конф. з паралельних та розподілених комп'ютерних систем 13–14 бер. 2013 р., Харків, Україна. – <http://hpc-ua.org/pdcs-13/files/proceedings/62.pdf>
30. Региональная сеть для науки и образования ChANT как инфраструктура для Грид-приложений / М.В. Григорьева, С.А. Крашаков, А.Ю. Меньшутин и др. // Материалы 4-й междунар. конф. ГРИД'2010 «Распределенные вычисления и Грид-технологии в науке и образовании», Дубна, 28 июня – 03 июля 2010 г. – С. 345–351.
31. AT&T:History of Network Management. – <http://www.corp.att.com/history/nethistory/management.html>
32. NOC Project site. – <http://kb.nocproject.org/display/SITE/About>
33. Introduction to Storage Area Networks and System Networking. – <http://www.redbooks.ibm.com/redbooks/pdfs/sg245470.pdf>
34. HP SAN Design Reference Guide. – <http://h20000.www2.hp.com/bc/docs/support/SupportManual/c00403562/c00403562.pdf>
35. Storage Networking Industry Association: «Common RAID Disk Data Format Specification». – [http://www.snia.org/sites/default/files/SNIA\\_DDF\\_Technical\\_Position\\_v2.0.pdf](http://www.snia.org/sites/default/files/SNIA_DDF_Technical_Position_v2.0.pdf)
36. OpenStack Cloud Administrator Guide. – <http://docs.openstack.org/admin-guide-cloud/admin-guide-cloud.pdf>
37. Уков Д. Блог компании Mirantis/OpenStack. – Хранение объектов для облака OpenStack: сравнение Swift и Ceph. – [http://habrahabr.ru/company/mirantis\\_openstack/blog/176195/](http://habrahabr.ru/company/mirantis/openstack/blog/176195/), 15 апр. 2013.
38. OpenStack Installation Guide for Ubuntu 12.04/14.04 (LTS). – <http://docs.openstack.org/icehouse/install-guide/install/apt/openstack-install-guide-apt-icehouse.pdf>
39. Heat Orchestration Template (HOT) Guide. – [http://docs.openstack.org/developer/heat/template\\_guide/hot\\_guide.html](http://docs.openstack.org/developer/heat/template_guide/hot_guide.html)
40. OpenStack High Availability Guide. – <http://docs.openstack.org/high-availability-guide/high-availability-guide.pdf>
41. HP представляет Converger Cloud // Сети и телекоммуникации. – 2013. – № 3. – С. 5.
42. CERN the European Organization for Nuclear Research. – <http://home.web.cern.ch/about>
43. OpenStack Users By Industry. – <http://www.openstack.org/user-stories/#Academic%20/%20Research%20/%20Government>
44. Bhanu P. Tholeti Hypervisors, virtualization, and the cloud... – [http://www.ibm.com/developerworks/cloud/library/cl-hypervisorcompare/index.html?S\\_TACT=105AGX99&S\\_CMP=CP](http://www.ibm.com/developerworks/cloud/library/cl-hypervisorcompare/index.html?S_TACT=105AGX99&S_CMP=CP)
45. OpenStack: Documentation. – <http://docs.openstack.org/>
46. Самойленко А. Виртуализация на платформах VMware Server и VMware ESX Server. – <http://www.ixbt.com/cm/vmware-server-esx-server.shtml>, апр. 2007.
47. Mell P., Grance T. The NIST Definition of Cloud Computing. National Institute of Standards and Technology // NIST Spec. Publ. 800-145. – <http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>, Sept. 2011.

Поступила 18.02.2015

Тел. для справок: +38 044 526-4159 (Київ)

E-mail: [aleksei@irtc.org.ua](mailto:aleksei@irtc.org.ua)

© В.И. Гриценко, А.А. Урсатьев, А.П. Лозинский, 2015