

УДК 004.6: 004.75

ОБЛАЧНЫЕ ТЕХНОЛОГИИ: ОСНОВНЫЕ ПОНЯТИЯ, ЗАДАЧИ И ТЕНДЕНЦИИ РАЗВИТИЯ

Т.В. Батура, к.ф.-м.н.; Ф.А. Мурзин, к.ф.-м.н.; Д.Ф. Семич, к.ф.-м.н.
(Институт систем информатики им. А.П. Ершова СО РАН, пр. Акад. Лаврентьева, 6,
г. Новосибирск, 630090, Россия, tbatura@ngs.ru, murzin@iis.nsk.su, deiman32@ngs.ru)

Аннотация. В статье делается обзор по облачным информационным технологиям. Под облачными вычислениями (от англ. *cloud computing*, также используется термин «облачная (рассеянная) обработка данных») обычно понимается предоставление пользователю компьютерных ресурсов и мощностей в виде интернет-сервисов. Рассмотрены основные модели предоставления услуг облачных вычислений: IaaS, PaaS и другие, особенности облачных платформ, хранилищ данных и программного обеспечения от различных поставщиков. Перечислены некоторые задачи, связанные с обработкой больших объемов данных, и обсуждаются тенденции развития.

Ключевые слова: облачные вычисления, *web-приложение*, виртуализация серверов, хранилище данных, центр данных.

Идея облачных вычислений появилась еще в 1960 году, когда Джон Маккарти высказал предположение, что когда-нибудь компьютерные вычисления будут производиться с помощью «общенародных утилит». Считается, что идеология облачных вычислений получила популярность с 2007 года благодаря быстрой развитию каналов связи и стремительному растущим потребностям пользователей.

Под облачными вычислениями (от англ. *cloud computing*, также используется термин «облачная (рассеянная) обработка данных») обычно понимается предоставление пользователю компьютерных ресурсов и мощностей в виде интернет-сервиса. Таким образом, вычислительные ресурсы предоставляются пользователю в «чистом» виде, и пользователь может не знать, какие компьютеры обрабатывают его запросы, под управлением какой операционной системы это происходит и т.д.

Часто облака сравнивают с мэйнфреймами (mainframe), находя между ними много общего. Принципиальное отличие облака от мэйнфреймов в том, что его вычислительная мощность теоретически не ограничена. Второе принципиальное отличие в том, что, попросту говоря, терминалы для мэйнфреймов служили только для интерактивного взаимодействия пользователя с запущенной на обработку задачей. В облаке же терминал сам является мощным вычислительным устройством, способным не только накапливать промежуточную информацию, но и непосредственно управлять глобальной системой вычислительных ресурсов.

Среди ранее возникших (в 1990-х гг.) технологий обработки данных некоторое распространение получили так называемые *grid*-вычисления. Это направление первоначально рассматривалось как возможность использования свободных ресурсов процессоров и развития системы добровольной аренды вычислительных мощностей. Ряд проектов (GIMPS, *distributed.net*, SETI@home) доказали, что такая модель вычислений достаточно эффективна. Сегодня эта технология применяется для решения научных, математических задач, где требуются значительные вычислительные ресурсы. Известно, что *grid*-вычисления также применяются для коммерческих целей. Например, с их помощью выполняются некоторые трудоемкие задачи, связанные с экономическим прогнозированием, анализом сейсмических данных, разработкой и изучением свойств вакцин и новых лекарств. Действительно, *grid*-вычисления и облака имеют много схожих черт в архитектуре и применяемых принципах. Тем не менее, модель облачных вычислений считается сегодня более перспективной благодаря значительно более гибкой платформе для работы с удаленными вычислительными ресурсами.

В настоящее время крупные вычислительные облака состоят из тысяч серверов, размещенных в *центрах обработки данных* (ЦОД). Они обеспечивают ресурсами десятки тысяч приложений, которые одновременно используют миллионы пользователей [1]. Облачные технологии являются удобным инструментом для предприятий, которым слишком дорого содержать собственные ERP, CRM или другие серверы, требующие приобретения и настройки дополнительного оборудования.

ERP (*Enterprise Resource Planning* – планирование ресурсов предприятия) – организационная стратегия интеграции производства и операций, управления трудовыми ресурсами, финансового менеджмента и управления активами, ориентированная на непрерывную балансировку и оптимизацию ресурсов предприятия посредством специализированного интегрированного пакета прикладного программного обеспечения, обеспечивающего общую модель данных и процессов для всех сфер деятельности предприятия.

CRM (Customer Relationship Management) – система управления взаимоотношениями с клиентами, то есть прикладное программное обеспечение, предназначенное для автоматизации стратегий взаимодействия с заказчиками (клиентами), в частности, для повышения уровня продаж, оптимизации маркетинга и

улучшения обслуживания клиентов путем сохранения информации о клиентах и истории взаимоотношений с ними, установления и улучшения бизнес-процедур и последующего анализа результатов.

Среди частных пользователей широкое распространение постепенно получают благодаря своему удобству такие облачные услуги, как, например, предоставляемые компанией Google («Документы», «Календарь» и др.).

Причины возрастающей популярности облачных технологий понятны: возможности их применения очень разнообразны и позволяют экономить как на обслуживании и персонале, так и на инфраструктуре. Аппаратное обеспечение может быть сильно упрощено при обработке данных и хранении информации в удаленных центрах данных. Все эти проблемы почти полностью переключаются на провайдера услуг.

К тому же такой подход позволяет стандартизировать ПО, даже если на компьютерах предприятия установлены разные операционные системы (Windows, Linux, MacOS и т.п.). Облачные технологии облегчают обеспечение доступа к данным компании как для клиентов, так и для собственных сотрудников, находящихся вне офиса, но имеющих возможность подключиться через Интернет.

Понятно, что использование облачных вычислений намного удобнее. Самым главным недостатком, который можно сразу заметить, является полная зависимость от поставщика этих услуг. Фактически предприятие (пользователь) оказывается заложником провайдера сервисов и провайдера доступа в сеть Интернет. Хотя надежность поставщиков облачных вычислений возрастает, для обеспечения надежности и безопасности данных необходимо приложить немало усилий, например, иметь дублирующие каналы связи, дублирующие мощности для возможности переключения на них и, конечно же, подумать о доступности информации и безопасности. Кроме этого, облачные вычисления совершенно не подходят для предприятий, имеющих отношение к государственной и военной тайне. Ни одна комиссия не выдаст сертификат на такую систему при работе с информацией, не подлежащей разглашению.

Как отмечено в [2], современные облачные технологии не только используются в готовом сетевом и серверном оборудовании, но и постепенно проникают на рынок встраиваемых систем (embedded cloud) и становятся причиной масштабной реструктуризации рынка. Внедрение встраиваемых систем приводит к размещению компьютерных процессоров в таких изделиях, как счетчики учета расхода ресурсов, интеллектуальные датчики, M2M-модули, автомобили, бытовая техника и т.д. Это позволяет управлять работой устройств, сбором данных и обеспечением интерактивных возможностей посредством подключения к компьютерной сети.

Идею подключения всевозможных устройств к глобальной сети называют *Интернетом вещей* (Internet of Things – IoT). По мнению Кевина Далласа, генерального менеджера Microsoft Windows Embedded [3], идея Интернета вещей существует уже много лет, однако для ее реализации не хватало одного звена, чтобы построить такую сеть, – облака.

Так как количество встраиваемых компьютеров увеличивается благодаря снижению цен на процессоры и повсеместному распространению Интернета, растут также и объемы передаваемых данных с последующей их обработкой (часто в режиме реального времени). Поэтому можно предположить, что в ближайшие годы роль Интернета вещей и облачных вычислений будет увеличиваться.

Основные модели предоставления услуг облачных вычислений

Модели развертывания облачных технологий

По модели развертывания облака разделяют на частные, общедоступные (публичные) и гибридные [2, 4].

Частные облака – это внутренние облачные инфраструктура и службы предприятия. Эти облака находятся в пределах корпоративной сети. Организация может управлять частным облаком самостоятельно или поручить эту задачу внешнему подрядчику. Инфраструктура может размещаться либо в помещениях заказчика, либо у внешнего оператора, либо частично у заказчика и частично у оператора. Идеальный вариант частного облака – облако, развернутое на территории организации, обслуживаемое и контролируемое ее сотрудниками.

Частные облака обладают теми же преимуществами, что и общедоступные, но с одной важной особенностью: предприятие само занимается установкой и поддержкой облака. Сложность и стоимость создания внутреннего облака могут быть очень высоки, а расходы на его эксплуатацию могут превышать стоимость использования общедоступных облаков.

Следует отметить, что у частных облаков есть преимущества перед общедоступными: более детальный контроль над различными ресурсами облака обеспечивает компании любые доступные варианты конфигурации. Кроме того, частные облака идеальны, когда нужно выполнять работы, которые нельзя доверить общедоступному облаку из соображений безопасности.

Общедоступные (публичные) облака – это облачные услуги, предоставляемые поставщиком. Они находятся за пределами корпоративной сети. Пользователи данных облаков не имеют возможности

управлять данным облаком или обслуживать его, вся ответственность возложена на владельца этого облака. Поставщик облачных услуг принимает на себя обязанности по установке, управлению, предоставлению и обслуживанию программного обеспечения, инфраструктуры приложений или физической инфраструктуры. Клиенты платят только за ресурсы, которые они используют.

Абонентом предлагаемых сервисов может стать любая компания и индивидуальный пользователь. Они предлагают легкий и доступный по цене способ развертывания веб-сайтов или бизнес-систем с большими возможностями масштабирования, которые в других решениях были бы недоступны. Примеры: онлайн-сервисы Amazon EC2 и Amazon Simple Storage Service (S3), Google Apps/Docs, Salesforce.com, Microsoft Office Web.

Вместе с тем услуги публичных облаков в основном предоставляются в виде стандартных конфигураций, то есть исходя из условий наиболее распространенных случаев использования. Это значит, что у пользователя остается меньше возможностей по выбору конфигурации по сравнению с системами, в которых ресурсами управляет сам потребитель. Следует также иметь в виду, что, поскольку потребители слабо контролируют инфраструктуру, процессы, требующие строгих мер безопасности и соответствия нормативным требованиям, не всегда подходят для реализации в общедоступном облаке.

Гибридные облака представляют собой сочетание общедоступных и частных облаков. Обычно они создаются предприятием, а обязанности по управлению ими распределяются между предприятием и поставщиком общедоступного облака. Гибридное облако предоставляет услуги, часть которых относится к общедоступным, а часть – к частным. Обычно такой тип облаков используется, когда организация имеет сезонные периоды активности. Другими словами, как только внутренняя ИТ-инфраструктура не справляется с текущими задачами, часть мощностей перебрасывается на публичное облако (например, большие объемы статистической информации, которые в необработанном виде не представляют ценности для предприятия), а также для предоставления доступа пользователям к ресурсам предприятия (к частному облаку) через публичное облако. Хорошо продуманное гибридное облако может обслуживать как требующие безопасности критически важные процессы, такие как получение платежей от клиентов, так и более второстепенные.

Основным недостатком этого типа облака является сложность эффективного создания подобных решений и управления ими. Необходимо получать услуги из разных источников и организовать их так, как если бы это был единый источник. Взаимодействие между частным и общедоступным компонентами может еще больше усложнить решение. Поскольку это относительно новая архитектурная концепция в сфере облачных вычислений, для этой модели появляются все новые и новые практические рекомендации и инструменты, и ее широкое распространение может затянуться до тех пор, пока она не будет лучше изучена.

По мнению Тома Биттмана, вице-президента и ведущего аналитика американской исследовательской и консалтинговой компании “Gartner” [5, 6], среди вышеперечисленных трех моделей развертывания облаков наиболее актуальной для бизнеса в данный момент являются частные облака. Биттман выделил пять основных моментов, которые помогают получить более точное представление об устройстве частного облака.

Облако – это не только виртуализация. Хотя виртуализация серверов и инфраструктуры составляет важный фундамент частных облачных вычислений, сами по себе виртуализация и управление виртуализированной средой еще не являются частным облаком.

Виртуализация позволяет лучше структурировать, объединять в пул и динамически предоставлять ресурсы инфраструктуры: серверы, десктопы, емкости для хранения, сетевое оборудование, связующее ПО и т.д. Но, чтобы среда технически могла считаться облачной, нужны еще и другие составляющие, такие как виртуальные машины, операционные системы или контейнеры связующего ПО, высокоустойчивые операционные системы, ПО grid-вычислений, ПО для абстрагирования ресурсов хранения, средства масштабирования и кластеризации.

Термин «частное облако» в отличие от общедоступного или гибридного относится к ресурсам, используемым единственной организацией, либо означает, что облачные ресурсы организации полностью изолированы в облаке от остальных.

Облако – обязательно источник экономии. Одно из главных заблуждений состоит в том, что облако будет экономить деньги. Экономия возможна, но не является обязательным атрибутом.

Частное облако позволяет более эффективно перераспределять ресурсы, чтобы удовлетворить корпоративные требования, и способно уменьшить капитальные затраты на оборудование. Но частное облако требует инвестиций в автоматизацию, и одна лишь экономия может не окупать всей стоимости. Так что, снижение затрат не является главным преимуществом этой модели. С этой точки зрения, главным стимулом к внедрению облачной модели должна быть не экономия, а скорость выхода на рынок, возможность быстрой адаптации и динамического масштабирования в соответствии со спросом, которые позволяют повысить скорость внедрения новых сервисов.

Частное облако не всегда внедрено у заказчика. Частное облако означает конфиденциальность, а не конкретное местоположение, владение ресурсами или самостоятельное управление. Многие поставщики предлагают нелокальные частные облака, то есть выделяют ресурсы единственному заказчику, исключая совместное использование одного пула несколькими клиентами. «Облако называется частным по его приватности, а не по тому, где оно развернуто, кто им владеет и несет ответственность за управление», – подчеркивает Биттман. Некоторые, например, могут свои ЦОД размещать у хостинг-провайдеров или объединять в пул ресурсы разных заказчиков, но изолировать их друг от друга с помощью виртуальной частной сети (Virtual Private Network – VPN) и других подобных технологий.

Частное облако (как и публичное облако) – это не только инфраструктурные сервисы. Серверная виртуализация – крупная тенденция и поэтому мощный двигатель частных облачных вычислений. Но частное облако не сводится только к инфраструктуре как услуге (IaaS). Например, для разработки и тестирования нового ПО высокоуровневая платформа как услуга (PaaS) имеет больше смысла, чем просто предоставление виртуальных машин.

Сегодня самый быстро растущий сегмент облачных вычислений – это IaaS. Она предоставляет самые низкоуровневые ресурсы ЦОД в простой для использования форме, но не меняет фундаментально принципы работы. Чтобы создать новые приложения, изначально предназначенные для облака и предоставляющие совершенно новые услуги, которые могут очень отличаться от того, что давали прежние приложения, разработчикам удобнее использовать PaaS.

Частное облако может перестать быть частным. С одной стороны, частное облако предоставляет преимущества облака: быстроту перестройки, масштабируемость и эффективность, избавляет от некоторых угроз безопасности, потенциальных и реальных, которые характерны для общедоступных облаков. С другой стороны, со временем уровень обслуживания, безопасность и контроль соблюдения требований в общедоступных облачных сервисах безусловно будут повышаться. Поэтому некоторые частные облака, возможно, целиком перейдут в категорию общедоступных. Большинство же сервисов частного облака, скорее всего, будут эволюционировать в гибридные облачные сервисы, расширяя доступные возможности за счет использования общедоступных облачных услуг и других сторонних ресурсов.

Основные свойства облачных технологий

Национальный Институт стандартов и технологий NIST (National Institute of Standards and Technology, USA) в своем документе “The NIST Definition of Cloud Computing” [5] определяет следующие характеристики облаков:

- возможность в высокой степени автоматизированного самообслуживания системы со стороны провайдера;
- наличие системы Broad Network Access;
- сосредоточенность ресурсов на отдельных площадках для их эффективного распределения;
- быстрая масштабируемость (ресурсы могут неограниченно выделяться и высвобождаться с большой скоростью в зависимости от потребностей);
- управляемый сервис (система управления облаком автоматически контролирует и оптимизирует выделение ресурсов).

Самообслуживание по требованию (On-demand self-service). У потребителя есть возможность получить доступ к предоставляемым вычислительным ресурсам в одностороннем порядке по мере потребности, автоматически, без необходимости взаимодействия с сотрудниками каждого поставщика услуг.

Широкий сетевой доступ (Broad network access). Предоставляемые вычислительные ресурсы доступны по сети через стандартные механизмы для различных платформ, тонких и толстых клиентов (мобильных телефонов, планшетов, ноутбуков, рабочих станций и т.п.).

Объединение ресурсов в пулы (Resource pooling). Вычислительные ресурсы провайдера объединяются в пулы для обслуживания многих потребителей по многоарендной (multi-tenant) модели. Пулы включают в себя различные физические и виртуальные ресурсы, которые могут быть динамически назначены и переназначены в соответствии с потребительскими запросами. Нет необходимости в том, чтобы потребитель знал точное местоположение ресурсов, однако можно указать их местонахождение на более высоком уровне абстракции (например, страна, регион или ЦОД). Примерами такого рода ресурсов могут быть системы хранения, вычислительные мощности, память, пропускная способность сети.

Мгновенная эластичность (Rapid elasticity). Ресурсы могут быть легко выделены и освобождены, в некоторых случаях автоматически, для быстрого масштабирования соразмерно спросу. Для потребителя возможности предоставления ресурсов видятся как неограниченные, то есть они могут быть присвоены в любом количестве и в любое время.

Измеряемый сервис (Measured service). Облачные системы автоматически управляют и оптимизируют ресурсы с помощью средств измерения, реализованных на уровне абстракции применительно для разного рода сервисов (например, управление внешней памятью, обработкой, полосой пропускания или

активными пользовательскими сессиями). Используемые ресурсы можно отслеживать и контролировать, что обеспечивает прозрачность как для поставщика, так и для потребителя, использующего сервис.

Модели обслуживания облачных технологий

В настоящее время принято выделять [4, 5, 7–9] три основные модели обслуживания облачных технологий, которые иногда называют слоями облака. Можно сказать, что эти три слоя – услуги инфраструктуры, услуги платформы и услуги приложений – отражают строение не только облачных технологий, но и информационных технологий в целом. Остановимся подробнее на каждом из них.

К **услугам инфраструктуры (Infrastructure as a Service – IaaS)** можно отнести набор физических ресурсов, таких как серверы, сетевое оборудование и накопители, предлагаемые заказчикам в качестве предоставляемых услуг. Услуги инфраструктуры решают задачу надлежащего оснащения ЦОД, предоставляя вычислительные мощности по мере необходимости. Обычно эти услуги поддерживают инфраструктуру и гораздо большее число потребителей по сравнению с услугами приложений. Частным примером услуг инфраструктуры является **аппаратное обеспечение как услуга (Hardware as a Service – HaaS)**. В качестве услуги пользователь получает оборудование, на основе которого разворачивает свою собственную инфраструктуру с использованием наиболее подходящего ПО.

Потребитель при этом не управляет базовой инфраструктурой облака, но имеет контроль над операционными системами, системами хранения, развернутыми приложениями и, возможно, ограниченный контроль выбора сетевых компонентов (например, хост с сетевыми экранами). В таком случае защиту платформ и приложений обеспечивает сам потребитель, а провайдер облака должен организовать защиту инфраструктуры. Для предоставления ресурсов по требованию часто используется виртуализация.

Преимущества. Снижение капиталовложений в аппаратное обеспечение. Поскольку в этой модели обычно используются методы виртуализации, можно добиться экономии в результате более эффективного использования ресурсов. Уменьшение риска потери инвестиций и порога внедрения, возможность плавного автоматического масштабирования.

Недостатки. Бизнес-эффективность и производительность очень зависят от возможностей поставщика. Существует вероятность, что потребуются потенциально большие долгосрочные расходы. Централизация требует новых подходов к мерам безопасности.

Примерами услуг инфраструктуры служат IBM SmartCloud Enterprise, VMWare, Amazon EC2, Windows Azure, Google Cloud Storage, Parallels Cloud Server и многие другие.

Услуги платформы (Platform as a Service – PaaS) – это модель обслуживания, в которой потребителю предоставляются приложения (созданные или приобретенные) как набор услуг. В него входят, в частности, промежуточное ПО как услуга, обмен сообщениями как услуга, интеграция как услуга, информация как услуга, связь как услуга и т.д. Например, **рабочее место как услуга (Workplace as a Service – WaaS)** позволяет компании использовать облачные вычисления для организации рабочих мест своих сотрудников, настроив и установив все необходимое для работы персонала ПО. **Данные как услуга (Data as a Service – DaaS)** предоставляют пользователю дисковое пространство, которое он может использовать для хранения больших объемов информации. **Безопасность как услуга (Security as a Service – SaaS)** дает возможность пользователям быстро разворачивать продукты, позволяющие обеспечить безопасное использование веб-технологий, безопасность электронной переписки, а также безопасность локальной системы. Этот сервис позволяет пользователям экономить на развертывании и поддержании своей собственной системы безопасности.

Другими словами, модель PaaS – это IaaS вместе с операционной системой и ее интерфейсом прикладного программирования (API – Application Programming Interface). Потребитель при этом не управляет базовой инфраструктурой облака, в том числе сетями, серверами, операционными системами и системами хранения данных, но имеет контроль над развернутыми приложениями и, возможно, некоторыми параметрами конфигурации среды хостинга. Таким образом, потребитель должен позаботиться об обеспечении защиты приложений, которые будут развернуты на предоставленных платформах.

Приложения могут работать как в облаке, так и в традиционных ЦОД предприятия. Для достижения масштабируемости, необходимой в облаке, различные предлагаемые услуги часто виртуализируются, как и рассмотренные ранее услуги инфраструктуры.

Преимущества. Плавное развертывание версий. Плавность означает, что в идеале пользователь должен слабо ощущать или даже вообще не ощущать изменения ПО в облаке.

Недостатки. Как и у предыдущей модели обслуживания, централизация требует надежных мер безопасности.

Примерами услуг платформы служат IBM SmartCloud Application Services, Amazon Web Services, Windows Azure, Boomi, Cast Iron, Google App Engine и другие.

Услуги приложений (Software as a Service – SaaS) предполагают доступ к приложениям как к сервису, то есть приложения провайдера запускаются в облаке и предоставляются пользователям по требо-

ванию как услуги. Другими словами, пользователь может получать доступ к ПО, развернутому на удаленных серверах, посредством Интернета, причем все вопросы обновления и лицензий на данное ПО регулируются поставщиком данной услуги. Оплата в данном случае осуществляется за фактическое использование ПО. Иногда эти услуги поставщики делают бесплатными, так как у них есть возможность получать доход, например, от рекламы.

Приложения доступны посредством различных клиентских устройств или через интерфейсы тонких клиентов, такие, например, как веб-браузер, или веб-почта, или интерфейсы программ. Потребитель при этом не управляет базовой инфраструктурой облака, в том числе сетями, серверами, операционными системами. На конечном пользователе лежит ответственность только за сохранность параметров доступа (логинов, паролей и т.д.) и выполнение рекомендаций провайдера по безопасным настройкам приложений.

Услуги приложений более всего знакомы повседневному пользователю. Самым распространенным примером приложений данного типа являются почтовые сервисы GMail, Mail.ru, Yahoo Mail. Вообще существуют тысячи приложений SaaS, и благодаря технологии Web 2.0 их число растет с каждым днем. Среди служб приложений имеется множество приложений, нацеленных на корпоративное сообщество. Существует ПО, управляющее начислением заработной платы, кадровыми ресурсами, коллективной работой, взаимоотношениями с клиентами и бизнес-партнерами и т.п.

Преимущества. Снижение капиталовложений в аппаратное обеспечение и трудовые ресурсы; уменьшение риска потери инвестиций; плавное итеративное обновление.

Недостатки. Как и в предыдущих двух моделях, централизация требует надежных мер безопасности.

Примерами SaaS являются Gmail, Google Docs, Netflix, Photoshop.com, Acrobat.com, Intuit QuickBooks Online, IBM LotusLive, Unyte, Salesforce.com, Sugar CRM и WebEx. Значительная часть растущего рынка мобильных приложений также является реализацией SaaS.

Существует мнение, что принятое в настоящее время деление облачных вычислений по мере развития технологий в ближайшем будущем уйдет в прошлое [2]. В облачных приложениях будущего, предположительно, будут сочетаться не только инфраструктурные и платформенные элементы от одного поставщика, но и различные сервисы, собранные от разных поставщиков. Возможно, в итоге облачные вычисления приведут к появлению концепции **Всё как услуга (Everything as a Service – EaaS)**. При таком виде сервиса пользователю будет предоставлено все – от программно-аппаратной части до управления бизнес-процессами, включая взаимодействие между пользователями.

Обзор решений ведущих вендоров

На сегодняшний день существует большое множество поставщиков облачных платформ, хранилищ и ПО. В связи с этим предпринимаются попытки каким-то образом сравнить их между собой. В Интернете можно найти массу обзоров на эту тему [10–12]. Но из-за многообразия предоставляемых услуг разобраться в том, кто из них лучше, довольно трудно. Поэтому необходимо выбрать несколько основных показателей, которые могут помочь при сравнении, например, емкость, защищенность, простота, функциональность, доступность и др. При этом, несомненно, каждый пользователь выбирает то или иное решение в зависимости от необходимости.

Сравнение платформ Amazon, Google и Microsoft

В настоящий момент основными поставщиками облачной инфраструктуры считаются Amazon, Google и Microsoft. У каждой из компаний имеется целая линейка предоставляемых услуг. В данных материалах описаны только некоторые из них, наиболее популярные. Также не обсуждается вопрос, к какой именно модели относится та или иная услуга и какие вендоры предоставляют только публичные облака, а какие могут участвовать в создании частных облаков.

Google [13]

<https://developers.google.com>

Google Drive – облачное хранилище данных, принадлежащее компании Google, позволяющее пользователям хранить свои данные на серверах в облаке и делиться ими с другими пользователями в Интернете. Google Drive отличается лаконичным интерфейсом и предлагает установить удобные программные клиенты для смартфонов и планшетов на базе операционной системы Android, ПК и ноутбуков под управлением операционной системы Windows или MacOS, мобильных устройств iPhone и iPad. В будущем ожидается более тесная интеграция хранилища с операционной системой Chrome OS и поддержка Linux. Каждый пользователь Google Drive получает до 15 Гбайт свободного пространства на все сервисы Google (в том числе Gmail и Photos). При этом он сам может решить, сколько места выделить под почту и какой объем оставить под важные файлы. Работать с файлами в Google Drive можно прямо в браузере.

Google Drive можно превратить в отдельную папку в документах смартфона, планшета или ПК, и ее содержимое будет синхронизироваться автоматически.

Google Docs – бесплатный онлайн-офис, включающий в себя текстовый, табличный процессоры и сервис для создания презентаций, а также интернет-сервис облачного хранения файлов с функциями файлообмена. Позволяет создавать и редактировать стандартные документы, таблицы и презентации, а также поддерживает функции совместной работы над ними.

Google App Engine – сервис хостинга сайтов и web-приложений на серверах Google. Бесплатно предоставляется до 1 Гб дискового пространства, 10 Гб входящего трафика в день, 10 Гб исходящего трафика в день, 200 миллионов гигабайт CPU в день и 2 000 операций отправления электронной почты в день. Приложения, разворачиваемые на базе App Engine, должны быть написаны на Python, Java либо Go. Предлагается набор API для сервисов хранилища datastore API (BigTable) аккаунтов Google, набор API для загрузки данных по URL, электронной почты и т.д.

Платформа Google конкурирует с аналогичными сервисами от Amazon, которые предоставляют возможность размещать файлы и веб-приложения, используя свою инфраструктуру. В отличие от многих обычных размещений приложений на виртуальных машинах, таких как Amazon EC2, платформа App Engine тесно интегрирована с приложениями и накладывает на разработчиков некоторые ограничения.

Google Cloud Storage – сервис хостинга файлов, основанный на IaaS. Все файлы, которые записываются или перезаписываются на серверы, автоматически шифруются по алгоритму AES-128. Является конкурентом продукта Amazon S3.

Amazon [14]

<http://aws.amazon.com>

Amazon Simple Storage Service (Amazon S3) – онлайн-веб-служба, предлагаемая Amazon Web Services, предоставляющая возможность для хранения и получения любого объема данных, в любое время из любой точки сети, так называемый файловый хостинг. В марте 2012 года компания Nasuni провела опыт, в течение которого поочередно передавала массивный объем данных (12 Тб) из одного облачного сервиса в другой [15]. В эксперименте участвовали наиболее рейтинговые облака: Amazon S3, Windows Azure и Rackspace. К удивлению исследователей, скорость передачи данных сильно отличалась в зависимости от того, какое облако принимало данные. Самый лучший показатель скорости записи данных оказался у Amazon S3, передача данных из двух других сервисов занимала всего 4–5 часов, в то время как передача данных в Rackspace заняла чуть меньше недели, а в Windows Azure – 40 часов.

Amazon Elastic Compute Cloud (Amazon EC2) – веб-сервис, предоставляющий вычислительные мощности в облаке. Он дает пользователям полный контроль над вычислительными ресурсами, а также доступную среду для работы. Amazon EC2 позволяет пользователям создать Amazon Machine Image (AMI), который будет содержать их приложения, библиотеки, данные и связанные с ними конфигурационные параметры, или использовать заранее настроенные шаблоны образов для работы Amazon S3. Amazon EC2 предоставляет инструменты для хранения AMI. Amazon S3 предоставляет безопасное, надежное и быстрое хранилище для хранения образов.

Microsoft [16]

<http://www.windowsazure.com>

Microsoft SkyDrive – интернет-сервис хранения файлов с функциями файлообмена, созданный и управляемый компанией Microsoft. Сервис SkyDrive позволяет хранить до 7 Гб информации (или 25 Гб для пользователей, имеющих право на бесплатное обновление) в виде стандартных папок. Пользователи могут просматривать, загружать, создавать, редактировать и обмениваться документами Microsoft Office (Word, Excel, PowerPoint и OneNote) непосредственно в веб-браузере. Присутствует удаленный доступ к компьютеру, работающему под управлением Windows.

Windows Azure – платформа облачных сервисов, разработанная Microsoft. Реализует модели PaaS и IaaS. Платформа предоставляет возможность разработки и выполнения приложений и хранения данных на серверах, расположенных в распределенных центрах данных.

- **Windows Azure Compute** – компонент, реализующий вычисления на платформе Windows Azure, предоставляет среду выполнения на основе ролевой модели.

- **Windows Azure Storage** – компонент хранилища, предоставляющий масштабируемое хранилище. Не имеет возможности использовать реляционную модель и является альтернативой (либо дополняющим решением) SQL Databases (SQL Azure) – масштабируемой «облачной» версией SQL Server.

- **Windows Azure Fabric** – по своему назначению является контролером и ядром платформы, выполняющая функции мониторинга в реальном времени, обеспечения отказоустойчивости, выделения мощностей, развертывания серверов, виртуальных машин и приложений, балансировки нагрузки и управления оборудованием.

Платформа Windows Azure имеет API, построенное на REST, HTTP и XML, что позволяет разработчикам использовать облачные сервисы с любой операционной системой, устройствами и платформами.

В работе [17] рассмотрены несколько типовых задач и сравниваются их возможные решения на каждой из платформ – **Amazon EC2, Google App Engine, Windows Azure**.

Задача 1. Необходимо отправить в облако на обработку локально созданное приложение; причем приложение, как правило, построенное на базе технологий Java или .NET, не должно менять свое содержание в процессе исполнения.

Задача 2. Необходимо запустить web-приложение, использующее фоновое обновление данных и средства балансировки нагрузки.

Задача 3. Необходимо выполнить в автоматическом режиме без применения интерфейса с пользователем прикладную вычислительную задачу с использованием методов параллельной обработки.

Задача 4. Необходимо запустить в облаке приложение, которое взаимодействует с локальной программой или обменивается данными с клиентским ПК.

Решения на платформе **Amazon**, как правило, базируются на том, что локальный компьютер рассматривается как сервер, входящий в состав общего ЦОД, и вносятся необходимые изменения в конфигурацию, либо необходимо создать несколько виртуальных машин в зависимости от требуемого уровня масштабирования.

На других вышеупомянутых платформах используют специальные средства динамического масштабирования, скриптовые программы, множество различных API. Часть из них весьма удобны, в других случаях необходимо прилагать достаточно много усилий для существенной или даже почти полной переработки исходной программы.

Еще некоторые крупные поставщики облачных технологий

IBM SmartCloud [18]

<http://www.ibm.com/cloud-computing/us/en/products.html>

Облачное решение, предлагаемое компанией IBM, а именно IBM SmartCloud, реализует все три модели (IaaS, SaaS, PaaS) в рамках не только публичного, но частного и гибридного облаков. В его состав входит облачный сервис, ранее называемый IBM Lotus Live, предоставляющий бизнес-приложения по модели SaaS. Содержит полный набор интерактивных сервисов, которые предоставляют масштабируемые решения для организации защищенной системы электронной почты, проведения web-конференций и коллективной работы. Сервисы свободны от рекламы и не собирают информацию о клиенте, а также не являются потребительскими приложениями, нацеленными на бизнес-деятельность. С пользователей взимается ежемесячная плата. Элементы управления системой защиты, развернутые для Lotus Live, обеспечивают приватность и управляемый доступ к важной информации при выполнении бизнес-операций. Все клиентские взаимодействия кодируются устойчивыми алгоритмами шифрования и осуществляются по протоколу SSL для HTTP и через RC2 в протоколе системы мгновенного обмена сообщениями Lotus Sametime. Резервные копии системы шифруются.

Rackspace Cloud [19]

<http://www.rackspace.com/cloud/>

Платформа предлагает набор продуктов для автоматизации хостинга и облачных вычислений, реализуется модель PaaS. Объединяет в себе Cloud Files, Cloud Servers, Cloud Sites. Благодаря серверной виртуализации пользователи получают возможность развертывать сотни облачных серверов одновременно и создавать архитектуру, обеспечивающую высокую доступность. Является конкурентом Amazon Web Services.

Oracle Exalogic Elastic Cloud [20]

<http://www.oracle.com/us/solutions/cloud/overview/index.html>

Компания ORACLE работает над концепцией ПО как услуги на протяжении последних 10 лет. На сегодняшний день компания признана одним из ведущих поставщиков ПО, построенного по технологии облака, и работает с более чем 5,5 миллиона пользователей. Компания ORACLE предлагает выбор между моделями развертывания ПО как с использованием ее центров данных и основанными на подписке, так и с моделями развертывания ПО на территории компании заказчика.

Для большинства центров данных, переходящих на технологию частных облачных вычислений, ORACLE в качестве первого шага предлагает консолидацию вычислительных ресурсов и переход на разделяемые и масштабируемые платформы и инфраструктуру.

Серверное оборудование, выделенное под индивидуальные задачи middleware, БД и другие приложения, рассчитано на пиковую нагрузку и обладает зарезервированной мощностью, не используемой постоянно. Каждый сервер может включать практически несовместимые программные компоненты от разных поставщиков ПО, что в ряде случаев увеличивает расходы на поддержку и повышает управленческие затраты.

Переход на объединенную архитектуру облачных вычислений ORACLE со стандартизированными разделяемыми приложениями по требованию существенно снижает издержки. Консолидация вычисли-

тельных ресурсов может быть выполнена как на уровне IaaS, с использованием технологий виртуализации, так и на уровне PaaS, с помощью стандартизации и объединения на основе БД и/или middleware архитектуры.

Консолидация на уровне PaaS представляет большую ценность, потому что уменьшает разнородность ПО, объединяя его на основе стандартизированных программных интерфейсов. Консолидация на уровне IaaS может также обеспечить более высокую эффективность разделения аппаратных средств, но ничего не дает для уменьшения сложности поддержки ПО. Наиболее популярной является консолидация ПО на уровне БД ORACLE, частично объединяющей преимущества PaaS и IaaS.

Для облачных вычислений компания ORACLE предлагает две ключевые технологии: виртуализация и кластеризация серверов. Виртуализация позволяет легко развертывать новые приложения по требованию и является хорошим способом разделения аппаратных средств между задачами. Объединение в кластеры важно для повышения диверсификации ресурсов между приложениями, тем самым повышая их доступность и отказоустойчивость.

Платформа ORACLE PaaS

ORACLE PaaS является масштабируемой платформой, общей для всех облачных приложений как частных, так и общественных центров данных. Платформа ORACLE PaaS основана на БД ORACLE и приложениях Oracle Middleware. Она дает возможность различным организациям объединять существующее ПО с использованием общей архитектуры, позволяющей создавать новые приложения, использующие существующие возможности ПО для расширения спектра услуг, предоставляемых по требованию.

Платформа ORACLE PaaS предоставляет услуги БД по требованию, основанные на БД ORACLE и аппаратных комплексах Oracle Exadata, а также услуги ПО Middleware по требованию на основе Oracle WebLogic и Oracle Exalogic. Oracle Exadata – это специализированная машина БД, а Oracle Exalogic является машиной, оптимизированной для выполнения приложений Middleware, написанных на языке JAVA. Обе машины масштабируемы и отказоустойчивы. Они спроектированы и сконфигурированы для совместной работы.

Для разработки новых приложений программисты могут использовать знакомые среды проектирования, такие как JDeveloper, NetBeans и Eclipse, а также сетевые инструменты WebCenter Page Composer, BI Composer и BPM Composer. Для взаимной интеграции новых и разработанных ранее приложений в частных и общественных облаках компания ORACLE предлагает Oracle SOA Suite и Oracle BPM Suite, а также Oracle Data Integration и Oracle GoldenGate. Идентификация пользователей и распределение прав осуществляются с помощью Oracle Identity and Access Management. За взаимодействие пользователей с облаком отвечает программный комплекс Oracle WebCenter, который позволяет пользователям осуществлять совместную работу.

Инфраструктура ORACLE IaaS

Компания ORACLE предлагает набор элементов технологии «Инфраструктура по требованию» (IaaS), в том числе вычислительные серверы, услуги хранения и передачи информации, виртуализацию ПО, операционные системы и системы управления ПО.

Инфраструктура ORACLE IaaS включает серверы, основанные на технологии SPARC и x86, установленные в стойках и лезвиях (blades); технологии хранения на FLASH, дисковых массивах и лентах; варианты виртуализации Oracle VM для x86, Oracle VM для SPARC и Oracle Solaris Containers; операционные системы Oracle Solaris, Oracle Linux, и Oracle Enterprise Manager.

Гибкая инфраструктура ORACLE IaaS поддерживает объединение гетерогенных ресурсов, масштабируемость, быстрое развертывание и высокую доступность прикладного ПО, позволяя эффективно управлять общественными и частными IaaS.

Аппаратные комплексы Oracle Exadata

Аппаратно-программный комплекс, серийно выпускаемый корпорацией Oracle, позиционируется как кластер серверов приложений для организации частных облачных и «эластичных» вычислений. Поставляется как предварительно собранный телекоммуникационный шкаф из 42 юнит, наполненный серверным и сетевым оборудованием. Аппаратная часть комплекса состоит из одноюнитовых серверов на базе двух процессоров Intel Xeon с двумя твердотельными накопителями в каждом для операционной системы и свопинга, общей для всех серверов системы хранения данных, коммутаторов InfiniBand и Ethernet. В последних выпусках X3-2 устанавливаются восьмиядерные Sandy Bridge частотой 2,9 ГГц, в каждом узле установлено 256 Гбайт оперативной памяти. Заказчикам комплекса предоставляется выбор из двух 64-разрядных операционных систем, возможных к предустановке на узлы кластера: Oracle Linux или Solaris (операционная система, разработанная компанией Sun Microsystems, которая сейчас принадлежит корпорации Oracle), а с середины 2012 года доступна возможность установки на узлы гипервизора Oracle VM.

Аппаратные комплексы Oracle Exadata выпускаются в виде стандартных стоек для размещения в ЦОД. Они состоят из серверов, использующих процессоры Intel XEON, основанные на архитектуре x86 и x64. В Oracle Exadata используются серверы двух типов: хранения данных и обработки. В качестве моста

между серверами используются коммутаторы InfiniBand и Ethernet. Как и любая модульная система, Oracle Exadata обладает свойством масштабируемости. Аппаратные комплексы Exadata выпускаются в нескольких вариантах, заполненных серверами в зависимости от предполагаемой максимальной нагрузки.

Каждый аппаратный комплекс Oracle Exadata содержит предварительно установленное ПО Oracle Database с опцией Real Application Cluster, позволяющей нескольким физическим серверам работать с единым хранилищем как единая БД без программных модификаций прикладного ПО.

Архитектура Oracle Exadata основывается на принципах симметричного доступа со всех серверов обработки ко всем узлам хранения (симметричного параллелизма). Такой подход является компромиссом, так как ориентирован на OLTP- и OLAP-обработку данных одновременно. Поэтому решение компании ORACLE не является лидером по быстродействию в узкоспециализированных задачах, но рассчитано на широкий спектр применения.

Специалисты компании считают, что комплексы Oracle Exadata как нельзя лучше подходят под идеологию облачных вычислений, так как являются универсальными и не требуют специализированного ПО.

Основатель компании Salesforce.com Марк Бениофф считает, что аппаратно-программные комплексы принципиально не могут решить задачу масштабируемости для конечного заказчика по сравнению с инфраструктурой, предоставляемой как услуга, а также отмечает, что концепция таких комплексов означает фактический возврат к концепции мейнфрейма, утратившей актуальность еще в конце 1970-х годов. Несмотря на возможность балансировки нагрузки в рамках нескольких приложений нет возможности функционирования выше предела вычислительных возможностей комплекса, и конечный заказчик вынужден приобретать аппаратные мощности под пиковую нагрузку. Подобного рода критика характерна для концепции частных облачных вычислений в целом.

В 2010 году Oracle открыла собственное публичное облако **Oracle Cloud**, предоставляющее как технологическое ПО по модели PaaS, так и бизнес-приложения по модели SaaS.

Salesforce.com [21]

<http://www.salesforce.com>

Система управления взаимоотношениями с клиентами (CRM-система – Customer Relationship Management) предоставляется заказчикам исключительно под модели SaaS. Под наименованием **Force.com** компания предоставляет PaaS-платформу для самостоятельной разработки приложений, а под брендом **Database.com** – облачную систему управления БД. В зависимости от уровня подписки доступны различные технические возможности. Так, в бесплатной версии Force.com подписчики могут создать не более десяти сущностей, а в неограниченной версии с ценой \$75 на пользователя в месяц – до 2 000. Подписчики могут размещать разработанные приложения на платформе Force.com в специальном каталоге (AppExchange) и предлагать свои разработки другим заказчикам, в том числе на коммерческой основе.

В качестве системы управления БД платформа Force.com использует три реплицируемых кластера Oracle RAC из восьми узлов каждый. Кластеры расположены в трех удаленных друг от друга ЦОД. В одной схеме Oracle Database обрабатываются данные сразу нескольких компаний-подписчиков. Используется стабильная схема данных, предварительно подготовленная к расширению дополнительными объектами таким образом, что в одних и тех же таблицах хранятся данные различных подписчиков независимо от различия в специфических атрибутах объектов для различных подписчиков. Широко используется секционирование таблиц БД.

Parallels [22]

<http://www.parallels.com>

Предлагает целый ряд продуктов для автоматизации хостинга и облачных вычислений, основанных на IaaS. Parallels Cloud Server объединяет в себе Parallels Cloud Storage, Parallels Virtuozzo Containers и Parallels Hypervisor, позволяя значительно повысить надежность, производительность и рентабельность серверов. Parallels Cloud Storage – это гибкое, масштабируемое решение, помогающее увеличить доступность и производительность облачных серверов. Parallels Virtuozzo Containers – решение для серверной виртуализации, обеспечивающее высокий уровень плотности и производительности, а также довольно быстрое выделение ресурсов и динамическое изменение сервис-планов. Технология Parallels Hypervisor позволяет самостоятельно создавать виртуальные машины.

Возможность размещать множество клиентов на изолированных друг от друга виртуальных серверах на одном физическом сервере позволяет сервис-провайдерам оптимизировать ресурсы и повышать прибыль. PACI (Parallels Automation for Cloud Infrastructure) – решение, включающее все необходимое для предоставления облачных услуг на любой аппаратной платформе. Клиенты могут расширять, сужать и удалять облачные услуги, распределять нагрузку между несколькими виртуальными машинами или контейнерами и защищать свои данные с помощью брандмауэров. PACI обеспечивает автоматическую балансировку нагрузки и безопасность (включая брандмауэры) для всех виртуальных серверов.

Slidebar [23]www.parking.ru

Облачная инфраструктура, предоставляемая в аренду IaaS. Продвигается под брендом SlideBar. Масштабируемые в реальном времени виртуальные машины с почасовой оплатой за мощность, измеряемую количеством арендованных процессорных ядер и объемом предоставленного хранилища данных. Дополнительно клиент может заказать гарантированную мощность процессоров (с учетом всех используемых ядер). SlideBar построен на облачной платформе Parking Cloud. Платформа использует виртуализацию аппаратного обеспечения Microsoft Hyper-V R2 и кластеризацию для распределения нагрузки, повышения надежности и работоспособности системы. Физически кластеры SlideBar размещаются в трех ЦОД компании Parking.ru, находящихся в Москве. Все дата-центры имеют дублированные каналы связи системы энергообеспечения, климатические системы, системы пожаротушения и постоянную круглосуточную охрану. Серверы кластеров построены на платформе HP с процессорами Intel серии 5500 и используют разделяемое сетевое хранилище (SAN) Cluster Shared Volume. Связь с облаком предусматривает неограниченный трафик с пропускной способностью до 100 Мбит/сек. (гарантированно – 2 Мбит/сек.).

«Трастинфо»<http://www.tietorussia.ru>

Совместное предприятие финской компании Tieto и российской компании «Ай-Тек». Со стороны Tieto добавлены собственные услуги по управлению операциями и технологии ИТ-сервисов. К ним относятся управление серверами, предоставление вычислительных мощностей по требованию, услуги связи и обеспечения безопасности данных, средства объединенных коммуникаций и совместной работы, управление приложениями, средства консолидации и оптимизации.

Российское облако предполагается сделать частью глобальной облачной сети Tieto, которая включает Санкт-Петербург, Хельсинки, Стокгольм, Осло и Копенгаген. Заказчикам будут гарантированы международный уровень качества услуг и доступ к сервисам и ресурсам Microsoft, Cisco и EMC. ЦОД «Трастинфо» имеет общую площадь 3000 м², количество стоек – 800, среднее энергопотребление на стойку – 5 кВт, пропускная способность волоконно-оптических линий связи до 10 Гбит/сек.

Опыт удачного применения готовых облачных решений

Первые проекты по внедрению Windows Azure в крупных российских ИТ-компаниях компанией Microsoft были представлены журналистам [24]. Так, известный разработчик электронных словарей и систем оптического распознавания текста компания АБВУУ открыла OCR-сервис FineReader Online на платформе Windows Azure. Пользовательская аудитория FineReader Online составляет примерно 250 тыс. человек. Согласно оценке самой компании АБВУУ, миграция на облачную платформу Microsoft позволяет сократить расходы на поддержку сервиса в полтора раза. Платформу Windows Azure активно осваивает также компания «Медиалогия» – разработчик первой в России автоматической системы мониторинга и анализа СМИ в режиме реального времени. Она разработывает системы анализа в Интернете российских и зарубежных блогов и СМИ с последующим предоставлением своим заказчикам мониторинга по темам и ключевым словам. Ее заказчиками являются большинство федеральных и региональных органов власти, а также крупнейшие корпорации. Как подчеркнул Фарит Хуснояров, директор по развитию компании «Медиалогия», этот бизнес является высокотехнологичным и высококонкурентным, поскольку медиапространство непрерывно развивается и совершенствуется. Перед компанией стоят задачи постоянного вывода на рынок новых продуктов и доработка уже существующих. Например, «Медиалогия» разработала специализированное решение, предназначенное для мониторинга блогов, где, как известно, наблюдается наибольшая протестная активность, при этом требуется более высокая интенсивность обновления информации, ведь блоггеры во многих случаях значительно оперативнее журналистов. Данная система была полностью создана на базе Windows Azure, и, по словам Фарита Хусноярова, срок вывода этого решения на рынок был сокращен в 2–3 раза по сравнению с традиционными способами разработки. Помимо системы мониторинга блогов, на базе Windows Azure были созданы агрегатор зарубежных СМИ и другие решения «Медиалогии», выпущенные в текущем году.

В Санкт-Петербургском национальном исследовательском университете информационных технологий, механики и оптики под руководством Бухановского А.В. разрабатывается облачная платформа CLAVIRE для обработки данных больших объемов. В основном платформа используется для обработки большого объема данных, получаемых в процессе наблюдений, экспериментов и математического моделирования на основе численных методов или компьютерной графики. Далее приведены ключевые особенности платформы CLAVIRE, определяющие ее архитектуру и функциональные характеристики [25].

1. Имеется предметно-ориентированный язык EasyPackage, который позволяет описать используемые прикладные пакеты и источники данных, процедуру их вызова, передачу параметров и используемые форматы данных.

2. Имеется предметно-ориентированный язык EasyFlow, который позволяет описать сложное (ком-
позитное) приложение в форме workflow.

3. Описанное на предметно-ориентированных языках композитное приложение может быть запу-
щено в среде облачных вычислений, тем самым получая в автоматическом режиме доступ к интегриро-
ванным в нее ресурсам: вычислительным сервисам, источникам данных.

4. Динамическое планирование выполнения композитного приложения позволяет добиться улуч-
шения производительности с использованием детерминированных и стохастических параметрических
моделей времени выполнения.

5. Вычислительные ресурсы, различающиеся архитектурой, режимом доступа, набором доступных
сервисов, интегрируются в рамках композитного приложения и вызываются в автоматическом режиме.

6. Высокоуровневый графический интерфейс предоставляет пользователю возможность работы с
системой с использованием понятий соответствующей предметной области.

7. Использование экспертных знаний позволяет обеспечить интеллектуальную поддержку работы
пользователя на протяжении всего процесса использования системы.

8. Динамический доступ к интегрированным источникам данных и технологическим средствам ви-
зуализации позволяет реализовать средства визуализации, начиная от простого построения графиков до
интерактивных систем виртуальной реальности.

Популярный в Интернете сервис для хостинга слайдов **SlideShare** реализовал перекодирование не-
скольких миллионов накопленных документов из Flash к формату HTML5 при помощи виртуальных
серверов компании Amazon [26]. Причин для апгрейда три. Во-первых, презентации на HTML5 отобра-
жаются на всех устройствах, включая смартфоны/планшеты iPhone/iPad и Android, и на десктопе – и это
один и тот же файл. Таким образом, уменьшается объем данных на хостинге. Во-вторых, документы ста-
ли на 40 % компактнее и загружаются на 30 % быстрее. В-третьих, документы теперь индексируются по-
исковыми системами. Текст без проблем выделяется мышкой и копируется, что всегда было затрудни-
тельно с Flash.

Технологии HP сыграли ключевую роль в создании многих кинокартин компании **DreamWorks**,
включая серию фильмов «Шрек», ленты «Кот в сапогах», «Как приручить дракона», «Кунг-Фу Панда» и
«Кунг-Фу Панда 2», говорится в заявлении компании [27]. Чтобы обработать большие объемы данных,
необходимых для производства, например, фильма «Кот в сапогах», DreamWorks внедрила облачные
сервисы HP, что позволило избежать наращивания собственных технических ресурсов, на которое, со-
гласно оценкам, потребовались бы миллионы долларов. 8 млн часов рендеринга из затраченных в общей
сложности 63 млн часов были выполнены с помощью облачных сервисов HP. На долю рендеринга при-
шло 45 % от общего объема использования облачных сервисов киностудией. Также решения HP задей-
ствованы в построении масштабируемой 10-гигабитной сетевой среды DreamWorks, состоящей из гло-
бальной и локальной сетей. Это семейство HP Networking: коммутаторы серий HP 12508 и 5800, а также
системы HP Networking Intelligent Management Center и HP Intelligent Resilient Framework. Кроме того, в
DreamWorks используется сетевая система хранения HP X9000 IBRIX, которая позволяет наращивать
емкость в соответствии с нынешними и будущими потребностями киностудии.

Опыт компании Microsoft

В многочисленных докладах Фабрицио Гальярди [28], имеющих в сети Интернет (около 20 докла-
дов), четко выражена позиция компании Microsoft относительно дальнейшего развития облачных вычис-
лений и дается информация о ведущихся проектах.

1. Облачные вычисления станут мощным средством для вычислительной науки, удалят некоторые
традиционные препятствия, снизят стоимость доступа к массивным вычислениям и обработке данных.

2. Облачные вычисления дадут возможность простого использования вычислительных ресурсов для
людей, не являющихся учеными, и для отдаленных институтов в менее развитых странах.

3. Это хорошая платформа для сохранения и распространения научных данных.

4. Доступность информации для финансирующих структур и общественности, что было продемон-
стрировано в проекте, объединяющем ЕС и Бразилию.

5. Корпорация Microsoft Research доступна для советов и поддерживает экспериментальный доступ к
этой технологии.

6. Microsoft Research настроена на взаимодействие с заинтересованными учеными России.

Облака основываются на больших центрах по сбору данных. В таблице 1 показано местонахождение
наиболее важных центров данных некоторых компаний [29], а на рисунке приведена фотография центра
данных компании Microsoft в Чикаго.

Таблица 1

Компания	Google	Apple	Microsoft	Yahoo
Местонахождение	Lenoir, TN	Apple, NC	Chicago, IL	La Vista, NE

В [30] дана достаточно подробная информация о центрах данных компании Google. На момент написания публикации у Google было 19 центров в США, 12 в Европе, один в России, один в Южной Америке и 3 в Азии. В действительности Google иногда арендует место в центрах данных других компаний.



Специалисты компании Microsoft отмечают, что в настоящее время необходимо провести научные исследования и сделать детальный экономический анализ деятельности для малого центра данных (1 тысяча серверов) и для большого (100 тысяч серверов). Изображенный на фотографии центр данных имеет размер в 11,5 раза больше размера футбольного поля, и можно только догадываться, сколько в нем сосредоточено серверов.

Главные мотивы развития центров данных

1. Высокая экологическая совместимость с внешней средой.
2. Эффективное использование энергии.
3. Адаптивное управление системами.
4. Возможность сосредоточить до 100 тысяч серверов.
5. Включение аппаратных средств не позднее, чем через неделю после поставки.
6. Запуск ПО не позднее, чем через несколько часов.
7. Живучесть во время отключения энергии или бедствия.
8. Большой набор ПО и услуг.
9. Непрерывная коммуникация, безопасность, надежность, производительность.

Некоторые наиболее важные исследовательские центры Microsoft Research в Европе, в которых ведутся исследования по облачным вычислениям: Barcelona Supercomputing Centre, Объединенный центр Microsoft Research и INRIA, The Microsoft Research – University of Trento Centre for Computational and Systems Biology (CoSBI).

Компания Microsoft Research подписала большое количество соглашений с партнерами в Европе, Азии и Латинской Америке по развитию облачных вычислений. Например, в рамках Европейского проекта Virtual multidisciplinary EnviroNments USing Cloud infrastructures (VENUS-C) подписано Европейско-Бразильское соглашение Open Data and Cloud Computing e-Infrastructure for Biodiversity (EUBrazilOpenBio), направленное на изучение биомногообразия, в нем также участвует Microsoft Research. Проект VENUS-C стремится поддерживать пользовательские сообщества посредством развития легких в использовании услуг, чтобы стимулировать разработку успешных облачных приложений.

Обработка больших объемов данных и некоторые задачи

В настоящее время для моделирования данных в различных предметных областях широко используются графы, позволяющие отобразить взаимные отношения объектов. Постоянно увеличивающиеся объемы данных этих приложений приводят к необходимости использовать масштабируемые платформы и параллельные вычислительные архитектуры, позволяющие эффективно обрабатывать очень большие массивы данных, возникающих в связи с анализом графов. Облачные вычисления применяются для решения задач, связанных с графами, в ряде предметных областей: семантический поиск [31], социальные

сети, базы знаний, моделирование фотонных кристаллов [32], поиск последовательностей ДНК [33] и т.д. Вопросам эффективного хранения и обработки данных в системах такого рода посвящено немало работ [34–36]. Имеется также много проблем, прямо не связанных с графами. Рассмотрим некоторые задачи, возникающие перед разработчиками облачных сервисов и перед использующими их специалистами.

Задачи распределения и использования ресурсов

При организации вычислительных процессов в сетях с облачной инфраструктурой объектами являются виртуальные машины, сервисы, программы, наборы данных, заявки; позициями – вычислительные узлы, устройства памяти, места в очередях на исполнение. При этом рассматривается ряд количественных характеристик: интенсивность поступающих запросов и степень загрузки центральных устройств, интенсивность межмашинного взаимодействия через сетевые адаптеры и др. Могут также учитываться характеристики индикаторного типа, например, наличие необходимого пакета на данном вычислительном узле [37]. Далее возникают, вообще говоря, оптимизационные задачи на графах, связанные с распределением и использованием ресурсов и составлением расписаний. Например, необходимо решить заданную прикладную задачу и при этом минимизировать сумму трафика между всеми парами узлов облака [36]. Для решения возникающих задач используют различные, как правило, приближенные, методы: оптимизационные, основанные на теории игр, статистические, машинного обучения [38].

Модель вычислений MapReduce

MapReduce – модель распределенных вычислений, предложенная компанией Google, используемая для параллельных вычислений над очень большими объемами данных. Точнее, MapReduce – это фреймворк для организации вычислительных процессов на распределенных системах, содержащих большое количество компьютеров, называемых нодами. Работа MapReduce состоит из двух шагов: Map и Reduce. На Map-шаге происходит предварительная обработка входных данных. Для этого один из компьютеров, называемый главным узлом (master node), получает входные данные задачи, разделяет их на части и передает другим компьютерам, называемым рабочими узлами (worker node), для предварительной обработки. На Reduce-шаге происходит «свёртка» предварительно обработанных данных. Главный узел получает ответы от рабочих узлов и на их основе формирует результат, то есть решение задачи. В рамках данной парадигмы были реализованы алгоритм Дейкстры нахождения кратчайшего пути в графе, различные алгоритмы нахождения значимых вершин в графе, метод ближайшего соседа, алгоритм байесовской классификации и др. [39–42].

Защита облачной инфраструктуры

В облачной информационной среде возникают многочисленные проблемы информационной безопасности: распространение вредоносного ПО, его обнаружение, выявление ПО, не являющегося вредоносным, но содержащим в себе ошибки, которые могут привести к возникновению деструктивных процессов [43]. Для решения возникающих задач обычно используются существующие решения: антивирусное ПО, системы обнаружения вторжений, системы предотвращения вторжений.

Однако в силу большого количества вычислительных узлов и больших объемов информации, циркулирующей в среде, а также ввиду неоднородности (например многоплатформенности) среды все задачи существенно усложняются. Например, сигнатурные методики затруднительно применять для облачных вычислений в средах, использующих различные программно-аппаратные платформы. Даже в случае, когда вредоносные алгоритмы будут одинаковыми, их реализации на различных платформах будут различаться. Таким образом, приходится хранить целый набор сигнатур. Статистические методы требуют слишком много информации о вычислительных узлах и программных интерфейсах. В частности, для каждой платформы необходимо иметь свой набор статистических данных. То есть возникающие задачи в принципе решаемые, но значительно усложняются в связи с мультиплатформенностью.

Обеспечение надежности работы множества серверов

Очевидно, что в распределенной системе отдельные серверы могут выходить из строя, поэтому необходимо иметь средства, позволяющие восстанавливать потерянную информацию. Простое дублирование (реплицирование) данных – это далеко не лучший способ. Более удобно использовать коды, корректирующие ошибки.

Предположим, что в нашем распоряжении имеется n серверов, S_1, \dots, S_n , и на k -м сервере хранится слово w_k в бинарном алфавите. Считаем, что длины этих слов равны $|w_1| = \dots = |w_n| = m$. Далее обозначим w_{ki} i -ю букву в слове w_k .

Рассмотрим теперь слово $W_i = w_{1i} \dots w_{ni}$ длины n . Применим к нему код Хэмминга. При этом так называемые проверочные символы будем приписывать справа. Поэтому можно написать, что $K(W_i) = W_i c_{1i} \dots c_{ri}$, где c_{1i}, \dots, c_{ri} – проверочные символы.

Здесь используем дополнительные серверы S'_1, \dots, S'_r . При этом на сервере S'_r будем хранить слово $C_r = c_{r1} \dots c_{rm}$. Все сказанное иллюстрирует таблица 2.

Таблица 2

Сервер	S_1	S_2	...	S_n	S'_1	...	S'_r
Данные	w_{11}	w_{21}	.	w_{n1}	c_{11}	.	c_{r1}

	w_{1m}	w_{2m}		w_{nm}	c_{1m}		c_{rm}

Можно сказать, что код Хэмминга «работает» внутри каждой строки таблицы. Ясно, что выход из строя любого из $N = n + r$ серверов не приведет к потере данных. Мы можем их восстановить, действуя по каждой строке в соответствии с кодом Хэмминга. Очевидно, что для восстановления данных при выходе из строя большого количества серверов необходимо использовать коды, исправляющие большее количество ошибок. Выигрыш очевиден, так как для такого рода кодов число r мало по сравнению с числом n . Например, для кода Хэмминга $r \approx \log_2 n$. На основе этих идей в Институте систем информатики им. А.П. Ершова СО РАН (г. Новосибирск) Мигинским Д.С. ведутся работы по созданию хранилища данных, предназначенного для хранения больших объемов генетической и другой информации.

Гомоморфные коды

Гомоморфное шифрование – это форма шифрования, позволяющая производить определенные математические действия с зашифрованным текстом и получать зашифрованный результат, который соответствует результату операций, выполняемых с открытым текстом. Обычно рассматривают операции сложения и умножения. Частично гомоморфные криптосистемы – это такие криптосистемы, которые гомоморфны относительно только одной операции (сложения или умножения). Например, коды RSA и Эль-Гамала гомоморфны относительно операции умножения. В 2009 году Крейгом Джентри из компании IBM был предложен полностью гомоморфный код [44], то есть код, гомоморфный для операций умножения и сложения одновременно. Очевидно, что гомоморфные коды естественно использовать в облачных средах. Однако при повышении уровня безопасности размер зашифрованного текста стремительно растет, что затрудняет применение метода Крейга Джентри на практике. Тем не менее, компания IBM выпустила свободную криптографическую библиотеку HElib с поддержкой гомоморфного шифрования [45].

Идентификация спам-страниц

Наглядным примером использования крупномасштабного графа является задача идентификации спам-страниц в поисковых сервисах сети Интернет. Поскольку для определения страниц, содержащих спам, требуются большие вычислительные возможности, разумно пытаться решить данную задачу посредством облачных технологий. Например, в работе [46] проблема идентификации спама представлена как задача бинарной классификации с обучаемым классификатором. Поскольку разделение нормальных web-страниц и страниц, содержащих спам, является линейно неотделимой проблемой, в упоминаемой работе предлагается применять «алгоритм с мягким зазором для метода опорных векторов» (Soft margin SVM – Soft margin Support Vector Machines). Для реализации алгоритма была выбрана распределенная вычислительная платформа Hadoop, а точнее HDFS (распределенная файловая система) и Hadoop MapReduce (фреймворк для реализации MapReduce-вычислений). HDFS – Hadoop Distributed Filesystem – принадлежит Apache Software Foundation, имеет открытый исходный код, предоставляет пользователям прозрачную низкоуровневую распределенную инфраструктуру. Инфраструктура MapReduce разработана компанией Google и имеет открытый исходный код реализации. На основе принципов MapReduce построена Hadoop MapReduce.

В алгоритме Soft margin SVM целевая функция для линейного классификатора [46] имеет вид

$$D(w) = \frac{1}{l} \sum_{i=1}^l R(w * x_i, y_i) + \lambda w * w,$$

где λ – параметр; $\|w\|^2 = w * w$ – классификационный интервал.

К целевой функции предлагается добавить регуляризационный фактор

$$\gamma \sum \alpha_{ij} \theta(w * x_i, w * x_j),$$

где α_{ij} – вес ссылки со страницы i на страницу j ; θ – штрафная функция.

Причем эта штрафная функция связана со структурой ссылок между страницами. Если рассматривать страницы в качестве вершин графа, а связи между страницами в качестве ребер, можно учесть тот факт, что спам-страницы часто ссылаются на нормальные страницы в целях повышения доверия, в то время как нормальные страницы в большинстве своем не ссылаются на страницы со спамом. Поэтому в качестве штрафной функции в рассматриваемой работе была использована функция вида

$$\theta(s, t) = \max(0, |s - t|).$$

Процесс идентификации спам-страниц с использованием Hadoop MapReduce можно описать следующим образом.

На Map-шаге происходит предварительная обработка входных данных. Для этого один из компьютеров (называемый главным узлом – master node) получает входные данные задачи, разделяет их на части и передает другим компьютерам (рабочим узлам – worker node) для предварительной обработки. Так, на Map-шаге каждый документ разбивается на слова и возвращаются пары $\langle key, value \rangle$, где ключом является само слово, а значением имя документа. Если в документе одно и то же слово встречается несколько раз, после предварительной обработки данного документа будет столько же пар, сколько раз встретилось это слово. Все пары с одинаковым ключом объединяются и передаются на вход функции reduce.

На Reduce-шаге происходит свертка предварительно обработанных данных. Главный узел получает ответы от рабочих узлов и на их основе формирует результат – решение задачи, которая формулировалась изначально. В данном случае функция reduce складывает все пары с одинаковым ключом и таким образом получает общее количество вхождений конкретного слова во все документы коллекции.

Предобработка текста, вычисление веса векторов и этап классификации осуществляются на платформе Hadoop и адаптированы к инфраструктуре MapReduce для повышения эффективности при параллельных вычислениях.

Адаптированная векторная модель (VSM – Vector Space Model) позволяет представлять коллекции документов векторами из одного общего для всей коллекции векторного пространства.

Более формально алгоритм можно описать следующим образом. Текст можно представить в виде n -мерного вектора $d = (w_1, w_2, \dots, w_n)$, где w_i ($i = 1, 2, \dots, n$) обозначает вес i -го термина (слова). В работе [46] для вычисления веса термина в векторной модели применялся алгоритм TFIDF (Term Frequency–Inverse Document Frequency), описываемый формулой

$$w_{ij} = tf_{ij} * idf_i,$$

где tf_{ij} – нормализованная частота термина, то есть нормированное количество появлений слова в документе (Normalized Term Frequency); idf_i – логарифм от обратной частоты документа относительно данного слова, то есть величина, обратная количеству документов, которые содержат конкретное слово (Reverse Document Frequency).

Нормализованная частота термина tf_{ij} вычисляется по формуле

$$tf_{ij} = \frac{f_{ij}}{|d_j|},$$

где f_{ij} – количество вхождений термина t_i в документ d_j ; $|d_j|$ – общее количество слов в документе.

Обратная частота документа idf_i вычисляется по формуле

$$idf_i = \log \frac{N}{df_i},$$

где N – общее количество документов в коллекции; df_i – количество документов, содержащих слово t_i по крайней мере один раз.

В итоге процесс вычисления величин tf_{ij} и idf_i сводится к вычислениям больших объемов данных, поэтому было предложено адаптировать модель программирования MapReduce. Например, процесс нахождения величины f_{ij} можно записать в следующем виде:

Mapper: $\langle Label, contents \rangle \rightarrow \langle (Label, token), 1 \rangle$

Reducer: $\langle \langle (Label, token), list(1) \rangle \rightarrow \langle (Label, token), f_{ij} \rangle$.

Здесь *token* – каждое отдельно взятое слово документа; *Label* – имя просматриваемого документа; *contents* – текст просматриваемого документа.

На стадии классификации web-страниц происходит разделение файлов согласно их размерам, Mapper и Reducer завершают задачу параллельно. Алгоритм позволяет улучшить точность распознавания и сократить время отклика.

Поиск информации

Еще один яркий пример применения облачных технологий – организация поиска. Например, в работе [47] рассматривается задача нечеткого поиска по ключевым словам в зашифрованных данных.

Основная идея безопасного нечеткого поиска по ключевым словам заключается в использовании двух этапов: построение нечеткого множества ключевых слов и разработка эффективного и безопасного алгоритма поиска.

Построение нечеткого множества слов w_i на основе подстановки с фиксированным расстоянием d обозначим $S_{w_i,d} = \{S'_{w_i,0}, S'_{w_i,1}, \dots, S'_{w_i,d}\}$, где $S'_{w_i,\tau}$ означает набор слов w'_i с шаблоном τ . Например, для слова *CASTLE* при $d = 1$ нечеткое множество ключевых слов, построенное на основе подстановки, будет $S_{CASTLE,1} = \{CASTLE, *CASTLE, *ASTLE, C*ASTLE, C*STLE, \dots, CASTL*E, CASTL*, CASTLE*\}$. Точнее – речь идет об окрестности определенного радиуса данного слова. При этом рассматривается метрика Левенштейна. Шаблоны используются для краткого описания множества слов, входящих в окрестность.

В общем случае для заданного ключевого слова w_i с длиной l размер множества $S_{w_i,1}$ при прямом подходе (то есть при перечислении всех слов, входящих в данное множество) будет равен $(2l + 1) \times 26 + 1$, в то время как при подходе с использованием шаблонов получается $2l + 1 + 1$, что намного меньше. Предложенный алгоритм позволяет значительно сократить место для хранения индексов, которые требуются при поиске: вместо 30 Гб необходимо примерно 40 М.

Схему нечеткого поиска по ключевым словам можно описать следующим образом.

1. Для построения индекса слова w_i с расстоянием d владелец данных сначала строит нечеткое множество ключевых слов $S_{w_i,d}$, используя технику, основанную на подстановке. Затем вычисляет множество так называемых люков $\{T_{w'_i}\}$ для каждого $w'_i \in S_{w_i,d}$ с секретным ключом sk , известным только владельцу данных и авторизованному пользователю. Владелец шифрует данные. Таблица индексов и зашифрованные файлы с данными передаются в облачный сервер для хранения.

2. Поиском по (w, k) авторизованный пользователь вычисляет множество так называемых люков $\{T_{w'}\}_{w' \in S_{w,k}}$, где $S_{w,k}$ также получено при построении нечеткого множества на основе подстановки. Затем он отправляет $\{T_{w'}\}_{w' \in S_{w,k}}$ на сервер.

3. При получении поискового запроса в виде набора люков $\{T_{w'}\}_{w' \in S_{w,k}}$ сервер отыскивает их в таблице индексов и возвращает все возможные идентификаторы зашифрованных файлов согласно определению нечеткого ключевого слова. Пользователь дешифрует полученные результаты и извлекает интересующие его файлы.

В некоторых случаях точный поиск по ключевым словам в зашифрованных данных оказывается неэффективным и может не удовлетворять пользователей, поэтому в облачных технологиях целесообразно применять технологии, сохраняющие конфиденциальность ключевых слов. В связи с этим и возникает задача нечеткого поиска. В работе [47] предложена техника построения нечетких множеств ключевых слов, основанная на подстановке в шаблоны, которая может быть применена для эффективного удаленного хранения данных и эффективной схемы поиска зашифрованных данных в облаке.

Тенденции развития облачных технологий

Таким образом, облачные вычисления можно рассматривать как новый подход, который даст мощный импульс дальнейшему развитию информационных технологий и вычислительных наук. Отметим, что распределенные и параллельные вычисления в Европе и Америке были поддержаны в широких масштабах. Например, за последние 10 лет в Европе в распределенные и параллельные вычисления было вложено свыше 1 миллиарда евро. В настоящее время в Европе развивается проект VENUS-C, который финансируется, чтобы более детально выявить возможности использования облачных вычислений для исследований и в промышленности.

Предшественниками облачных вычислений принято считать многочисленные и широко известные технологии, такие как ресурсные вычисления, grid-вычисления, виртуализация, гипервизоры и многое другое [4, 48]. Сервис-ориентированная архитектура (Service-Oriented Architecture – SOA) также сыграла важную роль в развитии облачных вычислений. Облачные вычисления являются в некотором смысле расширением SOA-приложений. В последнее время с SOA наравне с Web 2.0 более тесно связывают технологию мэшапов (Mashup). С технической точки зрения мэшап – это web-приложение, которое объединяет в один интегрированный инструмент данные, полученные из нескольких источников. Известными примерами мэшапов являются web-сервисы, использующие картографические данные Google Maps.

Однако это далеко не единственное направление развития мэшапов. Дальнейшее совершенствование связано с методами получения агрегированной информации из сети – web-потоки (RSS, Atom) [17]. Много делается в развитии методов ускоренного разбора HTML-страниц, когда от задачи требуется не получение макета страницы, а выборка определенной информации, представленной там. Интересны эксперименты с мэшапами, которые проводятся с использованием API для доступа сразу к нескольким сайтам. В связи с этим часто упоминают об интегрированной выборке информации с сайтов Amazon, eBay, Flickr, Google, Microsoft, Yahoo, YouTube. Еще один пример – недавно открытые API для российского почтового портала Mail.ru. Подобные приложения позволяют активно использовать российский информационный контент с переходом к облачным моделям вычислений.

Основная цель предприятий и поставщиков, осваивающих облачные решения [4], заключается в том, чтобы обеспечить предприятию ИТ-инфраструктуру как услугу. Опыт, накопленный при интеграции и предоставлении корпоративных приложений как отдельных услуг, сегодня пытаются применять и при организации уровней инфраструктуры. Предполагается, что ПО и физическая инфраструктура, как и приложения в SOA, должны быть доступными для обнаружения, управляемыми и регулируемые. Естественно, возникает необходимость создания специальных стандартов, которые описывали бы, как обнаруживать, потреблять, администрировать и регулировать услуги. Открытые стандарты имеют ключевое значение для получения максимальной мощности и гибкости от использования облачных технологий [4]. И хотя в настоящее время разработка новых стандартов еще продолжается, часть новых технологий находится на стадии активного внедрения. Например, для взаимодействия с облачными сервисами сейчас действует стандарт, предусматривающий использование браузеров на стороне клиента, которые поддерживают технологию AJAX. Их работа в автономном режиме должна соответствовать открытой спецификации HTML5.

Сегодня существует масса технологий, обеспечивающих решения на основе облачных вычислений. Если несколько лет назад их было трудно реализовать из-за отсутствия всеобъемлющих, понятных инструментов, таких как средства для упаковки и развертывания приложения в облачной инфраструктуре или привязки к инфраструктуре поставщика облака, то сейчас существуют стандарты, которые призваны обеспечить всеобщую поддержку средств виртуализации. Открытым стандартом для работы в облачной среде является OVF (Open Virtual Machine Format). Он описывает предъявляемые к поставщикам виртуальных услуг требования по упаковке и развертыванию виртуальных объектов, передаваемых клиентам облачных сервисов. Важно отметить, что OVF не накладывает ограничений на выбор гипервизора или применение определенной процессорной архитектуры. Это делает его открытым для платформ от разных поставщиков, что нашло подтверждение в полной поддержке предварительной версии OVF, оказанной практически всеми основными поставщиками средств виртуализации: Dell, HP, IBM, Microsoft, VMWare и XenSource.

Облачный стандарт также не ограничивает выбор программных решений, которые могут применяться для работы. Современный набор решений называют LAMP (акроним от Linux, Apache HTTP Server, MySQL и Perl/PHP/Python); обмен данными выстраивается на технологиях XML и JSON (текстовый формат обмена данными, основанный на JavaScript), для работы с web-сервисами предусматривается использование REST (REpresentational State Transfer). Эта технология описывает подход, когда для работы с данными должен применяться достаточно узкий набор стандартных форматов. Тем самым строго ограничивается разнообразие методов взаимодействия между объектами и снижается сложность задействованных протоколов [17].

Важными являются стандарты, обеспечивающие работу прикладных программ: поддержку коммуникационных функций на базе HTTP и XMPP (Extensible Messaging and Presence Protocol, открытый коммуникационный протокол для ПО класса Middleware, основанный на XML), средства безопасности (OAuth, OpenID, SSL/TLS) и агрегирования при передаче данных (Atom).

Для разработки ПО и тестирования необходимо приобретать, разворачивать, настраивать и содержать среду для выполнения тестирования. Ясно, что для быстрого создания и размещения такой среды как раз подходят облачные вычисления. В этом случае для групп разработчиков упрощаются непрерывная интеграция кода и процесс обновления продукта. Группы тестирования могут уделять больше времени проверке качества программного продукта и меньше заниматься организацией процесса тестирования.

Для групп разработчиков, помимо готовой среды исполнения, облако дает еще одно преимущество: оно может предложить такую разновидность SaaS, как инструменты как услуги. Вследствие этого интегрированная среда разработки (IDE – Integrated Development Environment) и простые редакторы кода становятся хостируемыми программами, доступными любому разработчику в любое время. Это исключает потребность в локальных средах разработки и, соответственно, в лицензиях на каждую машину, что, несомненно, удобно.

Существует еще один фактор влияния облачных вычислений на разработчиков. Для того чтобы облегчить понимание и поддержку исходного кода, упростить взаимодействие между разработчиками при создании одного и то же программного обеспечения, создаются стандартные интерфейсы прикладного

программирования (API). Конечно, разработчики стремятся придерживаться стандартов, но в некоторых случаях нестандартные API дают определенную выгоду в производительности. В облаке же любые отклонения от стандартных API особенно опасны. Потребители знают, что получают услуги от поставщика облака, но могут не знать деталей реализации этих услуг. Таким образом, преимущество облачных технологий – в универсальности решения. Но вместе с тем это достоинство оборачивается некоторым неудобством для разработчиков.

Как уже отмечалось ранее, слабым местом облачных технологий является безопасность информации. До сих пор стандартизация в этой области отсутствует. Это привело к тому, что каждый разработчик облачной платформы выбирает сегодня собственную модель обеспечения безопасности. Например, для Amazon EC2 и Eucalyptus (решение Open Source с целью создания частного облака) применяются пары из сертификатов X.509 и персональных ключей для аутентификации; в Google App Engine используется предварительная аккредитация через Google Accounts.

В результате возникшей разнородности пока наблюдаются трудности при обеспечении взаимодействия решений, построенных в разных облачных платформах. Одновременно это сдерживает внедрение новых функций защиты. Необходимость решения этих вопросов послужила причиной для создания в 2010 году рабочей группы Open Cloud Computing Interface (OCCI) Working Group, которая занимается разработкой стандартов в этой области. Сейчас ее деятельность затрагивает также выпуск стандартизованного набора допустимых операций, а также предоставление полной спецификации на процедуры аутентификации с применением стандартных для HTTP механизмов и средств шифрования на базе криптографических протоколов SSL/TLS.

Таким образом, можно выделить четыре направления, которые необходимо развивать для обеспечения безопасности построения облачного ЦОД:

- безопасное хранение данных в облачных хранилищах;
- безопасное исполнение заданий;
- безопасная передача данных;
- безопасный доступ к данным.

Однако, как можно заметить, облачные технологии имеют больше преимуществ, чем недостатков. К тому же нельзя не учитывать современные тенденции развития ИТ-индустрии, в которых все больше просматривается необходимость обращаться к облачным вычислениям, несмотря на отставание систем защиты информации.

Создание новых стандартов, в том числе для обеспечения безопасности облачных технологий, на сегодняшний день является приоритетной задачей, а дальнейшее развитие облачных решений будет осуществляться вместе с возникновением новых, более надежных способов защиты данных.

Некоторые вызовы и риски

В настоящее время развитие информационных технологий идет очень быстрыми темпами, поэтому естественны попытки в той или иной мере спрогнозировать будущее. В частности, обсуждаются различные возникающие вызовы и риски (challenges and risks), с которыми может столкнуться общество. Авторы данной статьи также попытались сделать определенные выводы на эту тему. Конечно, в процессе развития с течением времени может оказаться, что какие-то из высказанных положений могут оказаться неверными.

Итак, по мнению авторов, в строительстве больших центров данных можно увидеть стремление некоторых транснациональных корпораций типа Microsoft, IBM, Google к овладению большими объемами информации. Развитие происходящих сегодня событий можно представить следующим образом.

Например, некая компания сообщает, что наряду с обычным ПО ею разработан и облачный вариант. В действительности обычному пользователю облачный вариант может быть не нужен, но его может использовать какая-то компания. Хотя рекламируются удобство для коллективного использования и безопасность хранения данных, данные становятся доступными для владельцев облаков, то есть для владельцев серверов.

Компания может заявить, что в дальнейшем будет поддерживать только облачный вариант программ. Конечно, можно поддерживать обычный вариант на своем компьютере, но через некоторое время компьютер выйдет из строя, так как изначально, при производстве, заложен ограниченный ресурс времени его использования. На новом компьютере «железо» будет так изменено, что обычный вариант программ (не облачный) там уже не будет работать, как не будут работать и еще множество программ других компаний. Таким образом, вам не останется ничего другого, как «отправиться в облака», то есть ваши данные станут доступными для посторонних. Конечно, вы можете их кодировать, но в облаках работают специалисты более высокого класса, способные почти все декодировать.

Центры данных позволяют аккумулировать разнообразную информацию: политическую, военную, экономическую, социологическую, технологическую и др., с помощью автоматизированных методов

вести одновременный диалог с большим количеством людей. Например, компания IBM для обсуждения ряда технологических проблем привлекла около 140 тысяч человек. На специальном сайте Innovation Jam 2008 обсуждались вопросы изменения бизнес-моделей, взаимодействия с потребителями, глобальной интеграции и защиты окружающей среды. Innovation Jam строится на разработанном в IBM инструментарии для Web 2.0. Пройдя регистрацию, участник попадает на регулируемый модераторами сайт, где в форме форумов и чатов идет обсуждение по заданным темам и отвечающим от них вопросам. Здесь же в реальном времени отображается подробная статистика по дискуссиям в мире и отдельных странах. Помощь модераторам в слежении за ходом разговоров и в их последующей классификации оказывает автоматизированное средство обработки текста e-Classifier. То есть, грубо говоря, теперь оказывается возможным осуществлять мозговые штурмы с участием 140 тысяч человек.

Отдельная тема – социальные сети. Социальная сеть представляет собой интерактивный многопользовательский веб-сайт, содержание (контент) которого наполняется самими участниками сети. Сайт представляет собой автоматизированную социальную среду, позволяющую общаться группе пользователей, объединенных общими интересами. К ним, в частности, относятся и тематические форумы, особенно отраслевые, активно развивающиеся в последнее время. Количество зарегистрированных пользователей только одной сети может превышать население целой страны.

В последнее время все больше и больше говорят о персональных данных в Интернете, об анонимности и других подобных вещах. Касается это и социальных сетей, в которых хранится много личных данных. Самым безобидным можно считать использование личных данных самими владельцами социальных сетей для развития своего бизнеса. Но этими данными могут воспользоваться и преступники, в том числе совершающие финансовые махинации в электронных сетях.

Отдельно стоит сказать о рекламных технологиях. Многие социальные сети предоставляют прямую возможность проведения рекламы, и данные технологии быстро развиваются, начиная от простого разделения по полу (как, например, у сети FaceBook) и заканчивая сложными системами слежения и анализа действий пользователя, на основе которых ему и будет показана реклама. Существуют целые схемы с применением методов психологического воздействия с целью получения секретной информации, как правило, коммерческого характера.

Также заслуживают внимания технологии коллективного создания энциклопедий типа Wikipedia. Некоторые компании создают внутренние закрытые системы типа Wikipedia, содержащие технологическую информацию. Облачные вычисления расширяют эти возможности.

Как уже было отмечено, в центрах данных могут сосредотачиваться персональные данные. Можно предположить, что через некоторое время станет реальностью оперативная установка координат почти любого человека на планете. Через пять–десять лет любой человек за относительно небольшую сумму сможет получить свои генетические данные. Появление этих данных обусловит новые проблемы, так как из них можно будет выяснить не только предрасположенность к определенным заболеваниям и какими лекарствами человека лечить, но и какими ядами или вирусами на него легко воздействовать.

Центры данных могут быть объектами, интересными для террористов и как средство получения информации, и как объект террористических акций. В случае военных конфликтов центры данных могут стать важными целями. Естественно, что военные и спецслужбы типа Агентства национальной безопасности США уже создают свои центры данных, в том числе глубоко законспирированные.

Отметим, что создание центров данных на данном этапе является крайне нерентабельным. Их стоимость может превышать миллиард долларов. Компенсируются затраты за счет других видов деятельности, например, предоставления рекламных услуг в Интернете. Объемы данных, циркулирующих в обществе, по-видимому, еще долго намного будут превышать технические возможности по их накоплению и обработке. Поэтому процесс овладения большими объемами информации будет происходить не так успешно, как хотелось бы некоторым.

Учитывая сказанное, можно сделать вывод, что развитие облачных вычислений, несмотря на вызовы и риски, все-таки является тенденцией. Поэтому целесообразно участвовать в описанных выше процессах для того, чтобы действительно получать доступ к новым возможностям, которые появляются в связи с развитием облачных вычислений и больших центров данных.

Литература

1. Miller R. Who Has the Most Web Servers? 2012. URL: <http://www.datacenterknowledge.com/archives/2009/05/14/whos-got-the-most-web-servers/> (дата обращения: 29.05.2014).
2. Медведев А. Облачные технологии: тенденции развития, примеры исполнения // Современные технологии автоматизации. 2013. № 2. С. 6–9.
3. Dallas K. The Internet of Things is Here. 2012. URL: <http://blogs.msdn.com/b/windows-embedded/archive/2013/09/06/the-internet-of-things-is-here.aspx> (дата обращения: 29.05.2014).

4. Amrhein D., Quint S. Cloud computing for the enterprise: Part 1: Capturing the cloud. 2012. URL: http://www.ibm.com/developerworks/websphere/techjournal/0904_amrhein/0904_amrhein.html (дата обращения: 29.05.2014).
5. Облачные вычисления (Cloud computing). 2012. URL: [http://www.tadviser.ru/index.php/Статья:Облачные_вычисления_\(Cloud_computing\)](http://www.tadviser.ru/index.php/Статья:Облачные_вычисления_(Cloud_computing)) (дата обращения: 29.05.2014).
6. Gartner призывает к правильному пониманию частного облака. 2012. URL: <http://www.crn.ru/news/detail.php?ID=73064> (дата обращения: 29.05.2014).
7. Орландо Д. Модели сервисов облачных вычислений: инфраструктура как сервис. 2012. URL: <http://www.ibm.com/developerworks/ru/library/cloudservices1iaas/> (дата обращения: 29.05.2014).
8. Орландо Д. Модели сервисов облачных вычислений: платформа как сервис. 2012. URL: <http://www.ibm.com/developerworks/ru/library/cl-cloudservices2paas/> (дата обращения: 29.05.2014).
9. Орландо Д. Модели сервисов облачных вычислений: программное обеспечение как сервис. 2012. URL: <http://www.ibm.com/developerworks/ru/library/cl-cloudservices3saas/> (дата обращения: 29.05.2014).
10. Шуклин А. Топ-6 облачных хранилищ данных. 2012. URL: <http://digit.ru/technology/20130731/403909541.html> (дата обращения: 29.05.2014).
11. Хики Э. Топ-20 вендоров облачного хранения данных и решений для ЦОДов. 2012. URL: <http://www.crn.ru/news/detail.php?ID=63597> (дата обращения: 29.05.2014).
12. Маккарти Д. 100 самых перспективных поставщиков облачных решений. 2012. URL: <http://www.crn.ru/numbers/spec-numbers/detail.php?ID=79648> (дата обращения: 29.05.2014).
13. Google App Engine. 2012. URL: http://en.wikipedia.org/wiki/Google_App_Engine (дата обращения: 29.05.2014).
14. Amazon S3. 2012. URL: http://ru.wikipedia.org/wiki/Amazon_S3 (дата обращения: 29.05.2014).
15. Не все облачные хранилища одинаковы. 2012. URL: <http://www.hi-lo.ru/news/survey-not-all-storage-clouds-are-alike> (дата обращения: 29.05.2014).
16. Windows Azure. 2012. URL: http://en.wikipedia.org/wiki/Windows_Azure (дата обращения: 29.05.2014).
17. Новиков И. Облачные вычисления: на пороге перемен. 2012. URL: http://www.pcmag.ru/solutions/sub_detail.php?ID=44441&SUB_PAGE=1 (дата обращения: 29.05.2014).
18. Surana A., Vellal D., Guru R. Introducing IBM LotusLive. 2012. URL: <http://www.ibm.com/developerworks/lotus/library/lotuslive-intro/index.html> (дата обращения: 29.05.2014).
19. Rackspace Cloud. 2012. URL: http://en.wikipedia.org/wiki/Rackspace_Cloud (дата обращения: 29.05.2014).
20. Exalogic. 2012. URL: <http://ru.wikipedia.org/wiki/Exalogic> (дата обращения: 29.05.2014).
21. Salesforce.com. 2012. URL: <http://ru.wikipedia.org/wiki/Salesforce.com> (дата обращения: 29.05.2014).
22. Parallels. 2012. URL: <http://ru.wikipedia.org/wiki/Parallels> (дата обращения: 29.05.2014).
23. Slidebar. 2012. URL: <http://ru.wikipedia.org/wiki/Slidebar> (дата обращения: 29.05.2014).
24. Рудницкий Г. Microsoft представила первых крупных российских клиентов Windows Azure. 2012. URL: <http://www.it-weekly.ru/market/business/29410.html> (дата обращения: 29.05.2014).
25. Васильев В.Н., Князьков К.В., Чуров Т.Н., Насонов Д.А., Марьин С.В., Ковальчук С.В., Бухановский А.В. CLAVIRE: облачная платформа для обработки данных больших объемов // Информационно-измерительные и управляющие системы. 2012. Т. 10. № 11. С. 7–16.
26. Dignan L. SlideShare dumps Flash goes HTML5: Developer resources vs. multiple apps. 2012. URL: <http://www.zdnet.com/blog/btl/slideshare-dumps-flash-goes-html5-developer-resources-vs-multiple-apps/59006> (дата обращения: 29.05.2014).
27. Tadviser. 2012. URL: <http://www.tadviser.ru/index.php/Новости> (дата обращения: 29.05.2014).
28. Fabrizio Gagliardi – Director for Research Connections at Microsoft Research. 2012. URL: <http://eai.eu/bio/fabrizio-gagliardi-director-research-connections-microsoft-research> (дата обращения: 29.05.2014).
29. Kumar S., Buyya G., Buyya R. Green Cloud computing and Environmental Sustainability // IEEE Xplore Digital Library, Cloud Computing and Distributed Systems (CLOUDS) Laboratory. Dept. of Comp. Sci. and Software Engineering, Univ. of Melbourne, Australia, 2012, 27 p.
30. New Datacenter Locations. 2008. URL: <http://royal.pingdom.com/2008/04/11/map-of-all-google-data-center-locations/> (дата обращения: 29.05.2014).
31. C-Y Sheu P., Shu Wang, Qi Wang, Ke Hao, Ray P. Semantic Computing, Cloud Computing, and Semantic Search Engine. IEEE Intl Conf. on Semantic Computing, 2009, pp. 654–657.
32. O’ Brien N.S., Johnston S.J., Hart E.E., Djidjeli K., Cox S.J. Exploiting Cloud Computing for Algorithm Development. IEEE Intl Conf. on Cyber-Enabled Distributed Computing and Knowledge Discovery, 2011, pp. 336–342.
33. Doddavula S.K., Saxena V. Implementation of a Secure Genome Sequence Search Platform on Public Cloud. Third IEEE Intl Conf. on Cloud Computing Technology and Science, 2011, pp. 205–212.

34. Papadopoulos A., Katsaros D. A-Tree: Distributed Indexing of Multidimensional Data for Cloud Computing Environments. Third IEEE Intl Conf. on Cloud Computing Technology and Science, 2011, pp. 407–414.
35. Weinman J. Mathematical Proof of the Inevitability of Cloud Computing. 2011. URL: http://www.JoeWeinman.com/Resources/Joe_Weinman_Inevitability_Of_Cloud.pdf (дата обращения: 29.05.2014).
36. Петров Д.Л. Алгоритмы миграции данных в высокомасштабируемых облачных системах хранения // Автореф. ... канд. дисс. С.-Петербургский гос. электротехнич. ун-т «ЛЭТИ» им. В.И. Ульянова (Ленина). СПб, 2011. 18 с.
37. Суханов В.И. Минимизация трафика в облачной инфраструктуре // Науч. журн. КубГАУ. 2012. № 78 (04). С. 1–10.
38. Fei Teng Management des données et ordonnancement des tâches sur architectures distribuées. Thèse, Grade de Docteur, École Centrale Paris et Manufactures, 2011, 188 p.
39. Haerberlen A., Ives Z. Graph algorithms in MapReduce. NETS 212: Scalable and Cloud Computing. Univ. of Pennsylvania, October 15, 2013, Presentation, 61 slides.
40. Redekopp M., Simmhan Y., Prasanna V.K. Performance Analysis of Vertex-centric Graph Algorithms on the Azure Cloud Platform. Workshop on High-Performance Computing meets Databases (HPCDB), 2011, Salt Lake City, UT, USA, 8 p.
41. Kannan R., Vempala S., Woodruff D.P. Nimble Algorithms for Cloud Computing. 2012. URL: <http://arxiv.org/abs/1304.3162> (дата обращения: 29.05.2014).
42. Ghoting A., Kambadur P., Pednault E., Kannan R. NIMBLE: A Toolkit for the Implementation of Parallel Data Mining and Machine Learning Algorithms on MapReduce. Proc. of 17th ACM SIGKDD Conf. on Knowledge Discovery and Data Mining (KDD 2011). San Diego, CA, USA, 2011, pp. 334–342.
43. Туманов Ю.М. Защита сред облачных вычислений путем верификации программного обеспечения на наличие деструктивных свойств // Автореф. канд. дисс., М.: Изд-во НИЯУ «МИФИ», 2012. 20 с.
44. Gentry C. Fully Homomorphic Encryption Using Ideal Lattices // Proc. of 41st Annual ACM Symposium on Theory of Computing (STOC' 09). ACM NY, NY, USA, 2009, pp. 169–178.
45. Github. 2012. URL: <https://github.com/shaih/HElib> (дата обращения: 29.05.2014).
46. Chen J., Xu Y., Li Y. Research about spam page identification based on cloud computing in search service // 4th IEEE Intl Conf. on Intelligent Human-Machine Systems and Cybernetics, 2012, pp. 77–80.
47. Li J., Wang Q., Wang C., Cao N., Ren K., Lou W. Fuzzy keyword search over encrypted data in cloud computing. Mini-Conf. IEEE INFOCOM, 2010, Digital Object Identifier 10.1109/INFOCOM.2010.5462196, 5 p.
48. Гребнев Е. Облака: от старых технологий к широким перспективам. 2012. URL: http://cloud.cnews.ru/reviews/index.shtml?2011/05/20/440918_1 (дата обращения: 29.05.2014).