

Облачные вычисления

Облачные вычисления (*cloud computing*) — информационно-технологическая концепция, подразумевающая обеспечение повсеместного и удобного сетевого доступа по требованию к общему пулу (pool) конфигурируемых вычислительных ресурсов (например, сетям передачи данных, серверам, устройствам хранения данных, приложениям и сервисам — как вместе, так и по отдельности), которые могут быть оперативно предоставлены и освобождены с минимальными эксплуатационными затратами или обращениями к провайдеру.

Потребители облачных вычислений могут значительно уменьшить расходы на инфраструктуру информационных технологий (в краткосрочном и среднесрочном планах) и гибко реагировать на изменения вычислительных потребностей, используя свойства вычислительной эластичности (*elastic computing*) облачных услуг.

История

Ранние концепции использования вычислительных ресурсов по принципу системы коммунального хозяйства относят к 1960-м годам (к Джону Маккарти или Джозефу Ликлайдеру).

Следующими шагами к концептуализации облачных вычислений считаются появление [CRM](#)-системы [Salesforce.com](#), предоставляемой по подписке в виде веб-сайта (1999) и начало предоставления услуг по доступу к вычислительным ресурсам через Интернет книжным магазином Amazon.com (2002). Развитие сервисов Amazon, фактически превратившейся благодаря этим услугам в технологическую компанию, привело к формулировке идеи вычислительной эластичности и запуску в августе 2006 года проекта под названием *Elastic Computing Cloud* ([Amazon EC2](#)). Практически одновременно с запуском ЕСС термины *cloud* и *cloud computing* прозвучали в одном из выступлений главы Google Эрика Шмидта, начиная с этого времени встречаются многочисленные упоминания облачных вычислений в СМИ, в публикациях специалистов по информационным технологиям, в научно-исследовательской среде. Отсылка к «облаку» использовалась как метафора, основанная на изображении Интернета на диаграмме компьютерной сети, или как образ сложной инфраструктуры, за которой скрываются все технические детали.

Запуск в 2009 году приложений Google Apps отмечается как следующий важный шаг к популяризации и осмыслению облачных вычислений. В 2009—2011 годы были сформулированы несколько важных обобщений представлений об облачных вычислениях, в частности, выдвинута модель частных облачных вычислений, актуальная для применения внутри организаций, выделены различные модели обслуживания (SaaS, PaaS, IaaS). В 2011 году Национальный институт стандартов и технологий сформировал определение, которое структурировало и зафиксировало все возникшие к этому времени трактовки и вариации относительно облачных вычислений в едином понятии.

Характеристики

[Национальным институтом стандартов и технологий США](#) зафиксированы следующие обязательные характеристики облачных вычислений^[9]:

- *Самообслуживание по требованию (self service on demand)* — потребитель самостоятельно определяет и изменяет вычислительные потребности, такие как серверное время, скорости доступа и обработки данных, объём хранимых данных без взаимодействия с представителем поставщика услуг;
- *Универсальный доступ по сети* — услуги доступны потребителям по сети передачи данных вне зависимости от используемого терминального устройства;
- *Объединение ресурсов (resource pooling)* — поставщик услуг объединяет ресурсы для обслуживания большого числа потребителей в единый пул для динамического перераспределения мощностей между потребителями в условиях постоянного изменения спроса на мощности; при этом потребители контролируют только основные параметры услуги (например, объём данных, скорость доступа), но фактическое распределение ресурсов, предоставляемых потребителю, осуществляет поставщик (в некоторых случаях потребители всё-таки могут управлять некоторыми физическими параметрами

перераспределения, например, указывать желаемый центр обработки данных из соображений географической близости);

- *Эластичность* — услуги могут быть предоставлены, расширены, сужены в любой момент времени, без дополнительных издержек на взаимодействие с поставщиком, как правило, в автоматическом режиме;
- *Учёт потребления* — поставщик услуг автоматически исчисляет потреблённые ресурсы на определённом уровне абстракции (например, объём хранимых данных, пропускная способность, количество пользователей, количество транзакций), и на основе этих данных оценивает объём предоставленных потребителям услуг.

С точки зрения поставщика, благодаря объединению ресурсов и непостоянному характеру потребления со стороны потребителей, облачные вычисления позволяют экономить на масштабах, используя меньшие аппаратные ресурсы, чем требовались бы при выделенных аппаратных мощностях для каждого потребителя, а за счёт автоматизации процедур модификации выделения ресурсов существенно снижаются затраты на абонентское обслуживание.

С точки зрения потребителя эти характеристики позволяют получить услуги с [высоким уровнем доступности](#) (*high availability*) и низкими рисками неработоспособности, обеспечить быстрое масштабирование вычислительной системы благодаря *эластичности* без необходимости создания, обслуживания и модернизации собственной аппаратной инфраструктуры.

Удобство и универсальность доступа обеспечивается широкой доступностью услуг и поддержкой различного класса терминальных устройств ([персональных компьютеров](#), [мобильных телефонов](#), [интернет-планшетов](#)).

Модели развёртывания

Частное облако

Частное облако (*private cloud*) — инфраструктура, предназначенная для использования одной организацией, включающей несколько потребителей (например, подразделений одной организации), возможно также клиентами и подрядчиками данной организации. Частное облако может находиться в собственности, управлении и эксплуатации как самой организации, так и третьей стороны (или какой-либо их комбинации), и оно может физически существовать как внутри, так и вне юрисдикции владельца.

Публичное облако

Публичное облако (*public cloud*) — инфраструктура, предназначенная для свободного использования широкой публикой. Публичное облако может находиться в собственности, управлении и эксплуатации коммерческих, научных и правительственных организаций (или какой-либо их комбинации). Публичное облако физически существует в юрисдикции владельца — поставщика услуг.

Общественное облако

Общественное облако (*community cloud*) — вид инфраструктуры, предназначенный для использования конкретным сообществом потребителей из организаций, имеющих общие задачи (например, миссии, требований безопасности, политики, и соответствия различным требованиям). Общественное облако может находиться в кооперативной (совместной) собственности, управлении и эксплуатации одной или более из организаций сообщества или третьей стороны (или какой-либо их комбинации), и оно может физически существовать как внутри, так и вне юрисдикции владельца.

Гибридное облако

Гибридное облако (*hybrid cloud*) — это комбинация из двух или более различных облачных инфраструктур (частных, публичных или общественных), остающихся уникальными объектами, но связанных между собой стандартизованными или частными технологиями передачи данных и приложений (например, кратковременное использование ресурсов публичных облаков для [балансировки нагрузки](#) между облаками).

Модели обслуживания

Программное обеспечение как услуга

Программное обеспечение как услуга (SaaS, Software-as-a-Service) — модель, в которой потребителю предоставляется возможность использования [прикладного программного обеспечения](#) провайдера, работающего в облачной инфраструктуре и доступного из различных клиентских устройств или посредством [тонкого клиента](#), например, из [браузера](#) (например, веб-почта) или посредством интерфейса программы. Контроль и управление основной физической и виртуальной инфраструктурой облака, в том числе сети, серверов, операционных систем, хранения, или даже индивидуальных возможностей приложения (за исключением ограниченного набора пользовательских настроек конфигурации приложения) осуществляется облачным провайдером.

Платформа как услуга

Платформа как услуга (PaaS, [англ. Platform-as-a-Service](#)) — модель, когда потребителю предоставляется возможность использования облачной инфраструктуры для размещения базового программного обеспечения для последующего размещения на нём новых или существующих приложений (собственных, разработанных на заказ или приобретённых тиражируемых приложений). В состав таких платформ входят инструментальные средства создания, тестирования и выполнения прикладного программного обеспечения — [системы управления базами данных](#), [связующее программное обеспечение](#), среды исполнения языков программирования — предоставляемые облачным провайдером.

Контроль и управление основной физической и виртуальной инфраструктурой облака, в том числе сети, серверов, операционных систем, хранения осуществляется облачным провайдером, за исключением разработанных или установленных приложений, а также, по возможности, параметров конфигурации среды (платформы).

Инфраструктура как услуга

Инфраструктура как услуга (IaaS, [англ. Infrastructure-as-a-Service](#)) предоставляется как возможность использования облачной инфраструктуры для самостоятельного управления ресурсами обработки, хранения, сетями и другими фундаментальными вычислительными ресурсами, например, потребитель может устанавливать и запускать произвольное программное обеспечение, которое может включать в себя [операционные системы](#), платформенное и прикладное программное обеспечение. Потребитель может контролировать операционные системы, виртуальные системы хранения данных и установленные приложения, а также обладать ограниченным контролем за набором доступных сетевых сервисов (например, [межсетевым экраном](#), [DNS](#)). Контроль и управление основной физической и виртуальной инфраструктурой облака, в том числе сети, серверов, типов используемых операционных систем, систем хранения осуществляется облачным провайдером.

Экономические аспекты



При использовании облачных вычислений потребители информационных технологий могут существенно снизить [капитальные расходы](#) — на построение [центров обработки данных](#), закупку серверного и сетевого оборудования, аппаратных и программных решений по обеспечению

непрерывности и работоспособности — так как эти расходы поглощаются провайдером облачных услуг. Кроме того, длительное время построения и ввода в эксплуатацию крупных объектов инфраструктуры информационных технологий и высокая их начальная стоимость ограничивают способность потребителей гибко реагировать на требования рынка, тогда как облачные технологии обеспечивают возможность практически мгновенно реагировать на увеличение спроса на вычислительные мощности.

При использовании облачных вычислений затраты потребителя смещаются в сторону операционных — таким образом классифицируются расходы на оплату услуг облачных провайдеров.

Для объяснения экономической составляющей облачных подходов к вычислениям часто используется аналогия с услугами [водо-](#) или [электроснабжения](#), предоставляемыми в развитых инфраструктурах по соответствующим коммунальным сетям, легкодоступными и оплачиваемыми по мере потребления, в сравнении с разработкой каждым потребителем собственного [водозабора](#) или монтажом собственной [электроустановки](#)^[6].

Технологии

Для обеспечения согласованной работы узлов вычислительной сети на стороне облачного провайдера используется специализированное [связующее программное обеспечение](#), обеспечивающее мониторинг состояния оборудования и программ, [балансировку нагрузки](#), обеспечение ресурсов для решения задачи.

Облачные сервисы являются комбинацией существующих технологических решений, которые взаимно интегрированы для обеспечения максимального автоматизма и минимизации участия человека в работе комплекса. Можно выделить основные блоки, которые в первую очередь отличают «облачный» сервис от классического.

- Портал самообслуживания – инструмент, посредством которого пользователь может заказать для себя заранее предопределенный сервис с потенциальным уточнением деталей конфигурации (например, в случае с IaaS, виртуальную машину, уточнив объем требуемой оперативной памяти, число процессорных ядер, размер дискового пространства и т.п.), изменить параметры ранее заказанного сервиса или отказаться от него
- Каталог сервисов – список доступных пользователю сервисов и связанные с каждым из сервисов шаблоны их создания, то есть правила, по которым средства автоматизации будут данный сервис конфигурировать на реальном оборудовании и программном обеспечении
- Оркестратор – механизм, выполняющий последовательность операций, определенных в шаблоне для каждого сервиса
- Система тарификации и выставления счетов (биллинга) – механизм, определяющий объем потребленных пользователем ресурсов и соотношение с пользователем соответствующих финансовых затрат.

Одним из основных (но не обязательных) решений для сглаживания неравномерности нагрузки на услуги является размещение слоя серверной [виртуализации](#) между слоем программных услуг и аппаратным обеспечением. В условиях виртуализации балансировка нагрузки может осуществляться посредством программного распределения виртуальных серверов по реальным, перенос виртуальных серверов происходит посредством [живой миграции](#).

Критика

Концепция облачных вычислений с публичной моделью подвергалась критике со стороны сообщества [свободного программного обеспечения](#) и, в частности, со стороны [Ричарда Столлмана](#), считавшего, что использование стороннего веб-приложения, детали реализации которого не известны пользователю, ничем не отличается от применения проприетарного программного обеспечения с точки зрения пользовательского контроля за информацией^[10]. Существует вероятность, что с повсеместным приходом этой технологии станет очевидной проблема создания неконтролируемых данных, когда информация, оставленная пользователем, будет храниться годами, либо без его ведома, либо он будет не в состоянии изменить какую-то её часть. Примером того могут служить сервисы [Google](#), где пользователь не в состоянии

удалить неиспользуемые им сервисы и даже удалить отдельные группы данных, созданные в некоторых из них ([FeedBurner](#), [Google Friend Connect](#) и, возможно, другие).

Кроме того, некоторые аналитики предполагали появление к 2010 году проблем с облачными вычислениями. Так, например, Марк Андерсон, руководитель отраслевого IT-издания [Strategic News Service](#), считал, что из-за значительного притока пользователей сервисов, использующих облачные вычисления (например, [Flickr](#) или [Amazon](#)), растёт стоимость ошибок и утечек информации с подобных ресурсов, а в 2010 году должны были произойти крупные «катастрофы типа выхода из строя, или катастрофы, связанные с безопасностью». Так, например, в 2009 году сервис для хранения закладок Magnolia потерял все свои данные. Тем не менее, многие эксперты придерживаются той точки зрения, что преимущества и удобства перевешивают возможные риски использования подобных сервисов^[11].

SaaS

SaaS (*software as a service* — программное обеспечение как услуга; также [англ. software on demand](#) — программное обеспечение по требованию) — одна из форм [облачных вычислений](#), модель обслуживания, при которой подписчикам предоставляется готовое [прикладное программное обеспечение](#), полностью обслуживаемое провайдером. Поставщик в этой модели самостоятельно управляет приложением, предоставляя заказчикам доступ к функциям с клиентских устройств, как правило через [мобильное приложение](#) или [веб-браузер](#).

Основное преимущество модели SaaS для потребителя услуги состоит в отсутствии [затрат](#), связанных с установкой, обновлением и поддержкой работоспособности [оборудования](#) и работающего на нём программного обеспечения.

В модели SaaS:

- приложение приспособлено для [удаленного использования](#);
- одним приложением пользуется несколько клиентов (приложение коммунально);
- оплата взимается либо в виде ежемесячной абонентской платы, либо на основе объёма операций;
- [техническая поддержка](#) приложения включена в оплату;
- модернизация и обновление приложения происходит оперативно и прозрачно для клиентов.

Как и во всех формах облачных вычислений, заказчики платят не за владение программным обеспечением как таковым, а за его [аренду](#) (то есть за его использование через мобильное приложение или [веб-интерфейс](#)). Таким образом, в отличие от классической схемы [лицензирования программного обеспечения](#), заказчик несет сравнительно небольшие периодические затраты, и ему не требуется инвестировать значительные средства в приобретение прикладной программы и необходимых программно-платформенных и аппаратных средств для его развёртывания, а затем поддерживать его работоспособность. Схема периодической оплаты предполагает, что если необходимость в программном обеспечении временно отсутствует, то заказчик может приостановить его использование и заморозить выплаты разработчику^[1].

С точки зрения разработчика некоторого [проприетарного программного обеспечения](#) модель SaaS позволяет эффективно бороться с [нелицензионным использованием программного обеспечения](#), поскольку программное обеспечение как таковое не попадает к конечным заказчикам. Кроме того, концепция SaaS часто позволяет уменьшить затраты на развёртывание и внедрение систем технической и консультационной поддержки продукта, хотя и не исключает их полностью.

История

Изначально вся компьютерная отрасль использовала [арендную бизнес-модель](#) — первые [компьютеры](#) стоили огромных денег и их вычислительные мощности сдавались

заказчикам. С другой стороны, такую аренду нельзя считать разновидностью SaaS, поскольку заказчики получали доступ к компьютерам напрямую, а не через глобальные телеком-сети.

Поскольку модель SaaS ориентирована на предоставление услуг посредством сети, её развитие непосредственно связано с развитием глобальных сетей. Первые компании, предлагавшие программное обеспечение как услугу, появились в западных странах в 1997—1999 годах^[2], а акроним SaaS вошел в широкое употребление в 2001 году^[3].

Ключевые характеристики

Программное обеспечение по требованию обладает следующими ключевыми признаками:

- доступ к программному обеспечению, разработанному в соответствии с моделью ПО как услуга, предоставляется удалённо по сетевым каналам и, как правило, через веб-интерфейс, кроме того, могут использоваться [тонкие клиенты](#) и [терминальный доступ](#);
- программное обеспечение развёртывается в [центре обработки данных](#) в виде единого программного ядра, с которым работают все заказчики;
- программное обеспечение предоставляется на условиях уплаты периодических арендных платежей;
- обслуживание и обновление программного обеспечения выполняется централизованно на стороне поставщика приложения, предоставляемого как услуга (SaaS);
- стоимость [технической поддержки](#) обычно включается в арендную плату.

Стоимость

Программное обеспечение по требованию предоставляется заказчику в аренду и всегда предполагает периодическую оплату. В качестве единицы тарификации обычно используются пользователи (при предоставлении [CRM](#)) или же число записей в [базе данных](#) (при предоставлении [HRM-системы](#)), реже — какие-то другие функциональные характеристики (например, количество определённых операций или трафик)^[4]. В некоторых случаях заказчикам предлагаются комбинированные модели, в рамках которых могут дополнительно оплачиваться расширенные функции (например, заказчик может платить за пользователей своих сервисов и за расширенное хранилище данных).

Контракт на аренду SaaS включает в себя не только оплату за использование ПО, но и оплату всех затрат, связанных с поддержкой его работоспособности, обновлением и защитой данных. Ряд поставщиков SaaS предлагает продвинутый вариант контракта на аренду — [SLA \(Service Level Agreement\)](#). В таких контрактах фиксируются параметры, связанные с работоспособностью ПО. Обычно это гарантии доступности ПО в процентах в течение года. Лучшие центры обработки данных способны гарантировать доступность ПО не менее 99,5 % времени за год.

В том случае, если программное обеспечение не требует первоначальной адаптации под потребности заказчика, первоначальный платёж за ПО может отсутствовать вовсе. Данное обстоятельство является важнейшим преимуществом модели SaaS над классическим лицензированием программного обеспечения, которое также, — если лицензия на ПО платная, — требует существенных начальных инвестиций на его закупку. Периодические арендные платежи можно сравнить со стоимостью технической поддержки — обычно они жёстко прописываются в договоре и потому являются предсказуемыми. Тем самым обеспечивается защита инвестиций заказчика в используемый программный продукт.

Отличия от других моделей

В синонимы терминов SoD и SaaS часто ошибочно заносят термины [Hosted Applications](#) и [Application Service Provider \(ASP\)](#), за которыми стоит другая концепция продвижения ПО. Ключевое отличие SaaS от ASP состоит в том, что в рамках модели SaaS заказчик покупает доступ к единому программному ядру, которым пользуются все заказчики. Поставщик SaaS обеспечивает централизованное развитие и полное обслуживание программного ядра. В рамках же модели ASP каждому заказчику предоставляется выделенная инсталляция программного обеспечения, развёрнутая на удалённом сайте поставщика ПО и не доступная сразу нескольким заказчикам.

Многие поставщики утверждают, что они предоставляют SaaS-решение, но используют этот термин весьма небрежно. Именно работа заказчиков с единым программным ядром и его централизованное обслуживание поставщиком SaaS-решения обеспечивает основные положительные свойства SaaS (см. ниже).

Факторы, способствующие продвижению SaaS

Ключевым фактором, объясняющим экономическую целесообразность SaaS, является «эффект масштаба» — провайдер SaaS обслуживает единое программное ядро, которым пользуются все клиенты, и потому тратит меньшее количество ресурсов в сравнении с управлением отдельными копиями программного обеспечения для каждого заказчика. Кроме того, использование единого программного ядра позволяет планировать вычислительные мощности и уменьшает пиковые нагрузки для отдельных заказчиков. Все это позволяет SaaS-провайдерам существенно снизить стоимость эксплуатации ПО. В результате стоимость услуг для конечного пользователя такого ПО становится ниже издержек, возникающих при использовании классической модели лицензирования (особенно если лицензирование платное).

Другим ключевым фактором является уровень обслуживания SaaS. SaaS-провайдер способен предложить уровень обслуживания и поддержки ПО в работоспособном состоянии, недоступный для внутренних IT-отделов компаний. Это особенно ярко проявляется в случае работы с SaaS-провайдером контракта по SLA-контракту.

В настоящее время можно выделить несколько главных факторов⁶¹, стимулирующих использование программного обеспечения по требованию заказчиками и развитие данных продуктов разработчиками.

Положительные факторы SaaS для заказчиков

- Не нужна установка ПО на рабочие места пользователей — доступ к ПО осуществляется через обычный Web-браузер (реже через специальную программу-клиент);
- Радикальное сокращение затрат на развёртывание системы в организации. Это расходы на аренду помещения, организацию дата-центра, оплату труда сотрудников и т. д.;
- Сокращение затрат на техническую поддержку и обновление развернутых систем (вплоть до их полного отсутствия);
- Скорость внедрения, обусловленная отсутствием затрат времени на разворачивание системы;
- Понятный [интерфейс](#) — большинство сотрудников уже привыкли к использованию веб-сервисов;
- Ясность и предсказуемость платежей, [защита инвестиций](#);
- Мультиплатформенность — пользователь не зависит от программно-аппаратной платформы, выбранной разработчиком;
- Возможность получить более высокий уровень обслуживания [ПО](#).

Положительные факторы SaaS для разработчиков

- Рост популярности веб-сервисов для конечных пользователей;
- Развитие веб-технологий, большие функциональные возможности веб-приложений и простота их реализации;
- Быстрые процессы внедрения и сравнительно низкие затраты ресурсов на обслуживание конкретного клиента;
- Лёгкое проникновение на глобальные рынки;
- Отсутствие проблем с [нелицензионным распространением ПО](#);
- В отличие от классической модели, SaaS-клиент привязывается к разработчику — он не может отказаться от услуг разработчика и продолжать использовать систему. Таким образом, обеспечивается защита инвестиций разработчика в процесс продаж;
- В долгосрочном периоде доходы от SaaS могут превысить доходы от продаж лицензий и оказания технической поддержки (даже с учётом расходов на хостинг и управление приложениями);

- Разработчик выбирает рабочую программно-аппаратную платформу из соображений её технико-экономической эффективности, а не из соображений её распространенности у возможных пользователей ПО.

Сдерживающие факторы

Наряду с факторами, которые побуждают заказчиков внедрять программное обеспечение по требованию, а разработчиков — инвестировать ресурсы в его создание, существует ряд сдерживающих факторов, ограничивающих использование данной модели.

Во-первых, концепция SaaS применима далеко не для всех функциональных классов систем. Поскольку основная экономия ресурсов SaaS-провайдера достигается за счёт масштаба, SaaS-модели неэффективны для систем, требующих глубокой индивидуализации (адаптации под каждого заказчика), а также инновационных и нишевых решений.

Во-вторых, многие заказчики опасаются применять SaaS из-за соображений безопасности и возможной [утечки информации](#) со стороны поставщика SaaS-услуг. Вопросы, связанные с безопасностью, ограничивают использование SaaS-модели в критически важных системах, в которых обрабатывается конфиденциальная информация. С другой стороны, ответственность за утечку информации со стороны разработчика обычно регламентируется соответствующими договорами, и вероятность такой утечки часто ниже чем при использовании собственных внутренних систем. В том числе этому способствует недоступность программно-аппаратного комплекса, на котором развёрнута система, сотрудникам компании.

Третий фактор-ограничитель — необходимость постоянно действующего подключения к Интернету. Некоторые SaaS-продукты компенсируют это наличием модулей для автономной работы.

Критика SaaS

[Ричард Столлман](#) характеризует технологию как эквивалент всеобщего шпионского ПО и большой «чёрной двери» (дают оператору сервера неправомерную власть над пользователем)^[7]

Platform as a service

Platform as a Service (PaaS, «платформа как услуга») — модель предоставления [облачных вычислений](#), при которой потребитель получает доступ к использованию информационно-технологических платформ: [операционных систем](#), [систем управления базами данных](#), [связующему программному обеспечению](#), средствам [разработки](#) и [тестирования](#), размещённым у облачного провайдера. В этой модели вся информационно-технологическая инфраструктура, включая [вычислительные сети](#), [серверы](#), [системы хранения](#), целиком управляется провайдером, провайдером же определяется набор доступных для потребителей видов платформ и набор управляемых параметров платформ, а потребителю предоставляется возможность использовать платформы, создавать их виртуальные экземпляры, устанавливать, разрабатывать, тестировать, эксплуатировать на них [прикладное программное обеспечение](#), при этом динамически изменяя количество потребляемых вычислительных ресурсов.

Провайдер облачной платформы может взимать плату с потребителей в зависимости от уровня потребления, тарификация возможна по времени работы приложений потребителя, по объёму обрабатываемых данных и количеству [транзакций](#) над ними, по [сетевому трафику](#). Провайдеры облачных платформ достигают экономического эффекта за счёт использования [виртуализации](#) и [экономии на масштабах](#), когда из множества потребителей в одно и то же время лишь часть из них активно использует вычислительные ресурсы, потребители — за счёт отказа от [капитальных вложений](#) в инфраструктуру и платформы, рассчитанных под пиковую мощность и непрофильных затрат на непосредственное обслуживание всего комплекса.

В 2011 году мировой рынок публичных PaaS оценён в сумму около \$700 млн^[1], в числе 10 крупнейших провайдеров указываются Amazon.com (Beanstalk), Salesforce.com (Force.com, Heroku, Database.com), LongJump, Microsoft (Windows Azure), IBM (Bluemix^[en]), Red Hat (OpenShift), VMWare (Cloud Foundry), Google (App Engine), CloudBees, Engine Yard^[2].

В 2012 году в OASIS предложен стандарт для прикладного программного интерфейса управления облачными платформами CAMP (англ. *cloud application management for platforms*)^[3], определяющий унифицированные форматы для команд программного управления облачными платформами (таких как запуск, приостановка, запрос на выделение ресурсов)^[4].