

image not found or type unknown



ASP(Поставщик услуг приложений) — это компания, которая предлагает физическим лицам или предприятиям доступ через Интернет к приложениям и связанным с ними службам, которые в противном случае должны были бы располагаться на своих персональных или корпоративных компьютерах. SaaS (программное обеспечение как услуга) — 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. Выбор данной темы был основан на собственном интересе к SaaS-сервисам и их возможному будущему. Данное направление получило бурное развитие в последние годы и по сей день является крайне актуальным. Далее будут по отдельности рассмотрены ASP сервисы и SaaS сервисы и произведено сравнение между ними.

## ASP

Потребность в ASP была обусловлена ростом затрат на специализированное программное обеспечение, которое значительно превысило ценовой диапазон малого и среднего бизнеса. Предполагается, что услуги ASP, как правило, станут важной альтернативой не только для небольших компаний с низким бюджетом для информационных технологий, но и для крупных компаний как формы аутсорсинга или поставщиком многих услуг для частных лиц. Кроме того, растущие сложности программного обеспечения привели к огромным затратам на распространение программного обеспечения конечным пользователям. Через ASP можно снизить сложность и затраты на такое программное обеспечение. Кроме того, вопросы модернизации ПО будут исключены, передав бремя на ASP для поддержания актуальности услуг, технической поддержки 24 x 7, физической и виртуальной безопасности и встроенной поддержки непрерывности доступа для бизнеса и гибкой работы. Клиентами для служб ASP являются предприятия, правительственные организации, некоммерческие организации и членские организации.

Существует несколько форм ASP-бизнеса. Это:

- Функциональный ASP обеспечивает единое приложение, например, для обработки кредитных карт оплаты или расписаний услуг;
- Вертикальный рынок ASP обеспечивает пакет решений для конкретного типа клиента, например, стоматологической практики;
- Предприятие ASP обеспечивает решения широкого спектра;
- Местный ASP обеспечивает малый бизнес, услугами в пределах ограниченной области.

Некоторые аналитики идентифицируют объем ASP как пятый тип. Это в основном специализированный ASP, который предлагает недорогое пакетное решение через собственный веб-сайт. PayPal был представителем этого типа, и их объем был одним из способов снизить удельную стоимость каждой транзакции. В дополнение к этим типам некоторые крупные многолинейные компании (такие как HP и IBM ) используют концепции ASP в качестве конкретной бизнес-модели, которая поддерживает некоторых конкретных клиентов.

Прикладное программное обеспечение находится в системе поставщика и доступно пользователям через веб-браузер с использованием HTML или специального программного обеспечения(клиента), предоставляемого поставщиком. Пользовательское клиентское программное обеспечение также может взаимодействовать с этими системами через XML API или даже REST API. Эти API также могут использоваться там, где требуется интеграция с внутренними системами. ASP могут использовать или не использовать многопользовательскую установку при развертывании программного обеспечения для клиентов; некоторые ASP предлагают экземпляр или лицензию для каждого клиента (например, с помощью виртуализации).

Общие функции ASP, включают:

- ASP полностью владеет и управляет программным приложением (приложениями)
- ASP владеет, управляет и обслуживает серверы, поддерживающие программное обеспечение
- ASP предоставляет доступную информацию клиентам через Интернет или «тонкий клиент»,
- ASP-счета выставляются «за использование» или имеют ежемесячную / годовую плату

К преимуществам такого подхода относятся:

- Проблемы с интеграцией программного обеспечения переходят с клиентской стороны на ASP
- Затраты на программное обеспечение распределяются на несколько клиентов
- Поставщики могут создавать больше специальных приложений, чем внутренние разработчики отдельных компаний
- Платформы разработки с низким содержанием кода позволяют упростить настройку предварительно созданных приложений
- Ключевые программные системы постоянно обновляются, доступны и управляются специалистами
- Повышенная надежность, доступность, масштабируемость и безопасность внутренних ИТ-систем
- Соглашение об уровне обслуживания поставщика гарантирует определенный уровень обслуживания
- Доступ к экспертам по продуктам и технологиям
- Снижение внутренних затрат на ИТ до предсказуемой ежемесячной платы
- Переключение ИТ-персонала и инструментов с поддержки на стратегические технологические проекты, которые влияют на итоговую прибыль предприятия

К числу присущих им недостатков относятся:

- Клиент обычно должен принимать приложение, как это предусмотрено производителем (без каких-либо доработок), поскольку ASP могут предоставлять индивидуальное решение только для крупнейших клиентов.
- Клиент может полагаться на поставщика, чтобы обеспечить критическую бизнес-функцию, тем самым ограничивая свой контроль над этой функцией
- Изменения на рынке ASP могут привести к изменению типа или уровня обслуживания, доступного клиентам
- Интеграция с системами, отличными от ASP, может быть проблематичной.

Оценка безопасности поставщика услуг приложений при переходе на инфраструктуру ASP может стоить очень дорого, так как такая фирма должна оценить уровень риска, связанный с самим ASP. Неспособность должным образом учитывать такой риск может привести к:

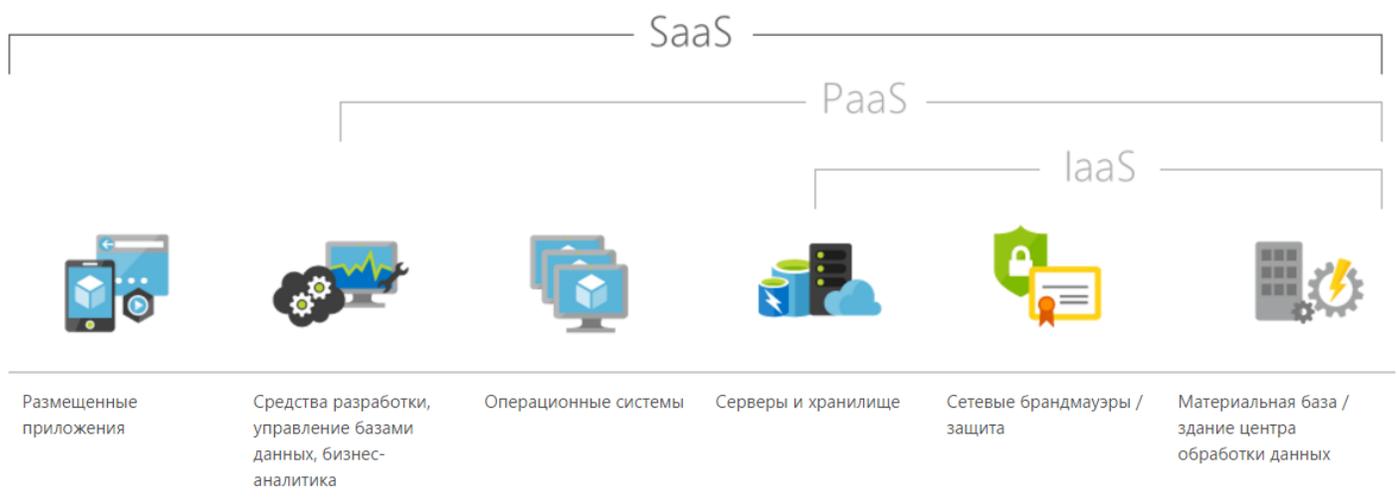
- Потере контроля над корпоративными данными
- Потере контроля над корпоративным имиджем
- Недостаточной безопасностью ASP для противодействия рискам
- Использованию корпоративных данных другими пользователями ASP

Физические посещения ASP для оценки формальности организации обеспечат бесценное понимание осведомленности о фирме.

## SAAS

SaaS (Software-as-a-Service - программное обеспечение как услуга)- является лицензией на программное обеспечение и моделью поставки, в которой программное обеспечение лицензируется на основе подписки и размещается централизованно. SaaS обычно доступен пользователям, использующим тонкий клиент через веб-браузер. SaaS стала общей моделью поставки для многих бизнес-приложений, включая ПО для офиса и обмена сообщениями, программное обеспечение для обработки платежных ведомостей, ПО для СУБД, программное обеспечение для менеджмента, программное обеспечение САПР, программное обеспечение для разработки, виртуализацию, учет, сотрудничество и управление отношениями с клиентами (CRM), информационные системы управления (MIS), планирование ресурсов предприятия (ERP), выставление счетов, управление человеческими ресурсами (HRM), управление контентом (CM) и управление службами поддержки. SaaS был включен в стратегию почти всех ведущих компаний программного обеспечения для предприятий.

Потребителем SaaS является конечный пользователь. В зону ответственности поставщика услуг SaaS входит поддержание доступности и работоспособности поставляемых приложений, предоставляя при этом минимальный набор изменяемых параметров.



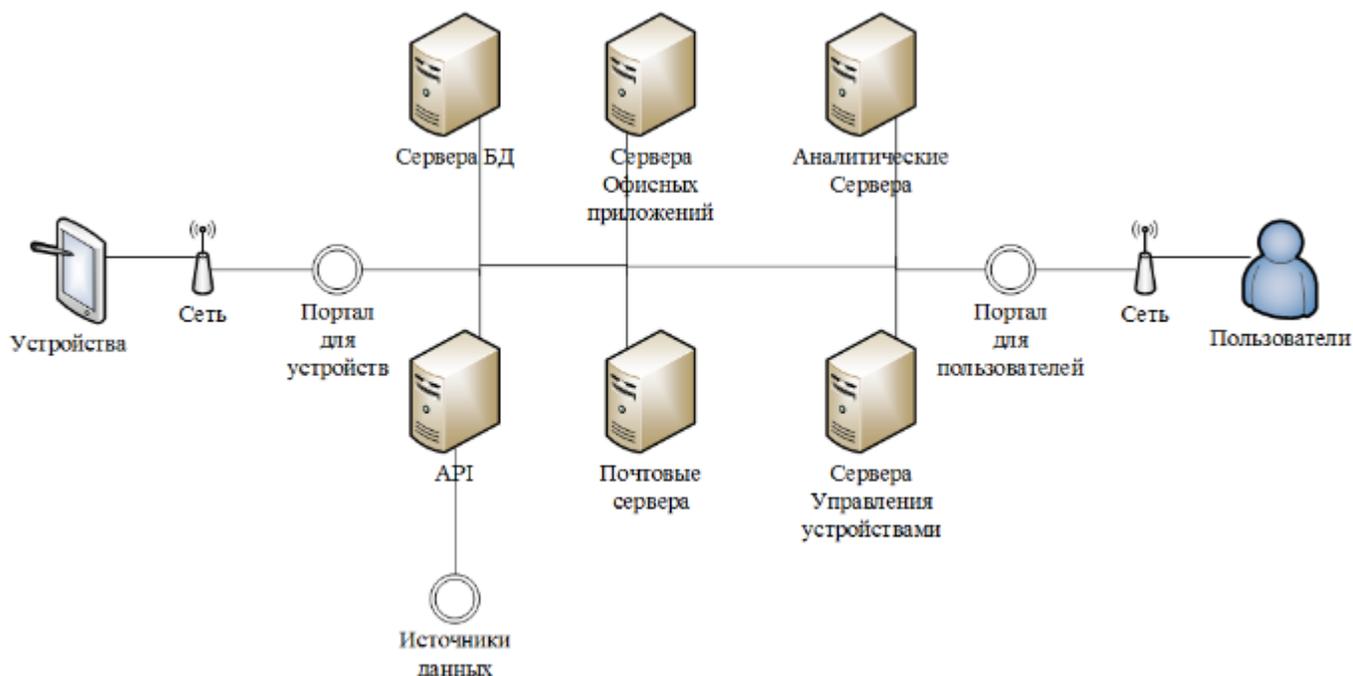
Термин "программное обеспечение как услуга" (SaaS) считается частью номенклатуры облачных вычислений наряду с инфраструктурой как услугой (IaaS), платформой как услугой (PaaS), настольным компьютером как услугой (DaaS),

управляемое программное обеспечение как услуга (MSaaS), мобильный сервер как услуга (MBaaS) и управление информационными технологиями как услуга (ITMaaS).

Для SaaS на стороне клиента необходимы лишь рабочие места - компьютеры и канал связи с облачным провайдером. Для потребления услуг SaaS на стороне клиента нет необходимости в содержании технического специалиста. Для стабильной работы будет достаточно менеджера с минимальными ИТ - навыками лишь для поддержания контакта с поставщиком услуг SaaS. SaaS не имеет физической необходимости (как и любое облако) для распространения, потому что она не распределяется физически и развертывается почти мгновенно, поэтому обычные в таком случае посредники и партнеры не нужны. В отличие от традиционного программного обеспечения, которое условно продается как бессрочная лицензия с единоразовой покупкой (и дополнительно постоянной поддержкой). SaaS - провайдеры обычно используют абонентскую плату, чаще всего ежемесячную, либо годовую. Следовательно, первоначальная стоимость установки для SaaS обычно ниже, чем эквивалентное корпоративное программное обеспечение. Поставщики SaaS обычно оценивают стоимость своих услуг на основе некоторых параметров, таких как количество пользователей, использующих приложение или за транзакцию, событие, а также за количество требуемых процессоров. Относительно низкая стоимость предоставления пользователю (то есть настройка нового клиента) в среде с несколькими клиентами позволяет некоторым поставщикам SaaS предлагать приложения с использованием модели freemium. В этой модели предоставляется бесплатный сервис с ограниченными функциональными возможностями или областью применения, а плата взимается за расширенные функциональные возможности или более широкий охват. Некоторые другие SaaS-приложения полностью бесплатны для пользователей, при этом доход получается из альтернативных источников, таких как реклама. Ключевым фактором роста SaaS является способность продавцов SaaS предоставлять цены, конкурентоспособные с локальным программным обеспечением.

Подавляющее большинство решений SaaS основаны на многопользовательской архитектуре. С этой моделью для всех клиентов («арендаторов») используется одна версия приложения с единой конфигурацией (аппаратное обеспечение, сеть, операционная система). Для поддержки масштабируемости приложение устанавливается на нескольких машинах (называемое горизонтальным масштабированием). В некоторых случаях вторая версия приложения настроена так, чтобы предлагать избранной группе клиентов, имеющим доступ к предварительно выпущенным версиям приложений (например, бета-версии) для

целей тестирования. Несмотря на исключение, а не на норму, некоторые решения SaaS не используют многопользовательскую работу или используют другие механизмы, такие как виртуализация, чтобы экономически эффективно управлять большим количеством клиентов вместо многоуровневости.



Существует два основных типа SaaS:

1. Вертикальные SaaS - Программное обеспечение, отвечающее потребностям конкретной отрасли (например, программное обеспечение для здравоохранения, сельского хозяйства, недвижимости, финансовых отраслей)
2. Горизонтальные SaaS - Продукты, которые ориентированы на категорию программного обеспечения (маркетинг, продажи, инструменты для разработчиков, HR).

Приложения SaaS также поддерживают то, что традиционно называют настройкой приложения. Другими словами, как и традиционное корпоративное программное обеспечение, один клиент может изменить набор параметров конфигурации (параметров), которые влияют на его функциональность и внешний вид. У каждого клиента могут быть свои собственные настройки (или: значения параметров) для параметров конфигурации. Приложение может быть настроено в той степени, в которой оно было разработано на основе набора predetermined параметров конфигурации.

Приложения SaaS обновляются чаще, чем традиционное программное обеспечение, во многих случаях еженедельно или ежемесячно. Это обеспечивается несколькими факторами:

- Приложение размещается централизованно, поэтому обновление определяется и выполняется поставщиком, а не клиентами.
- Приложение имеет только одну конфигурацию, что ускоряет тестирование разработки.
- Поставщику приложения не нужно тратить ресурсы на обновление и поддержку backdated версий программного обеспечения, потому что существует только одна версия.
- Поставщик приложения имеет доступ ко всем данным о клиентах, ускоряя проектирование и регрессионное тестирование.
- Поставщик решений имеет доступ к поведению пользователя в приложении (обычно через веб-аналитику), что упрощает определение областей, требующих улучшения.

Поскольку приложения SaaS не могут получить доступ к внутренним системам (базам данных или внутренним службам) компании, они преимущественно предлагают интеграционные протоколы и API, которые работают по глобальной сети. Как правило, это протоколы на основе HTTP, REST и SOAP. Вездесущность приложений SaaS и других интернет-сервисов и стандартизация их технологий API породили разработку гибридных приложений, которые представляют собой легкие приложения, которые объединяют данные, визуальную часть и функциональность из нескольких сервисов, создавая сложную службу.

Положительные факторы роста популярности SaaS:

1. Растущее использование прикладных интерфейсов на основе веб-приложений, а также распространение связанных с ними практик (например, веб-дизайн) постоянно уменьшали потребность в традиционных клиент-серверных приложениях. Следовательно, инвестиции традиционного поставщика программного обеспечения в программное обеспечение, основанное на толстых клиентах, стали недостатком (обязательным условием постоянной поддержки), открывая двери для новых поставщиков ПО, предлагающих пользователям более «современные» решения.
2. Стандартизация технологий веб-страниц (HTML, JavaScript, CSS), рост популярности веб-разработки как практики, а также внедрение и повсеместность таких веб-приложений, как Ruby on Rails или Laravel (PHP),

постепенно снизили стоимость разработки новых SaaS Решений и позволило новым поставщикам решений выработать конкурентные решения, бросая вызов традиционным поставщикам.

3. Увеличение проникновения широкополосного доступа в Интернет позволило удаленным централизованно размещенным приложениям предлагать скорость, сопоставимую с внутренним программным обеспечением.
4. Стандартизация протокола HTTPS как части веб-стека обеспечивала универсальную легкую безопасность, достаточную для большинства повседневных приложений.
5. Внедрение и широкое принятие легких интеграционных протоколов, таких как REST и SOAP, позволило обеспечить доступную интеграцию между приложениями SaaS (находящимися в облаке) с помощью внутренних приложений в глобальных сетях и с другими приложениями SaaS.

Некоторые ограничения замедляют развитие SaaS и запрещают его использование в некоторых случаях:

1. Поскольку данные хранятся на серверах поставщика, основной проблемой становится безопасность данных.
2. Приложения SaaS размещаются в облаке, вдали от пользователей приложения. Это вводит задержку в работу сервисов; Так, например, модель SaaS не подходит для приложений, которые требуют времени ответа в миллисекундах.
3. Многоуровневые архитектуры, которые обеспечивают экономическую эффективность для поставщиков решений SaaS, ограничивают настройку приложений для крупных клиентов, препятствуя использованию таких приложений в сценариях (применимых главным образом к крупным предприятиям), для которых такая настройка необходима.
4. Некоторые бизнес-приложения требуют доступа или интеграции с текущими данными клиента. Когда такие данные являются большими по объему или чувствительны (например, личная информация конечных пользователей), их интеграция с удаленно размещенным программным обеспечением может быть дорогостоящей или рискованной или может противоречить правилам управления данными.
5. Смена поставщиков SaaS может включать медленную и сложную задачу передачи очень больших файлов данных через Интернет.
6. Организации, которые используют SaaS, могут обнаружить, что они вынуждены использовать новые версии, что может привести к

непредвиденным затратам на обучение, увеличению вероятности того, что пользователь может сделать ошибку, или нестабильности от ошибок в более новом программном обеспечении.

7. Если поставщик программного обеспечения закрывает свой бизнес, пользователь может неожиданно потерять доступ к своему программному обеспечению, что может дестабилизировать текущие и будущие проекты его организации, а также оставить пользователю более старые данные.
8. Опора на подключение к Интернету означает, что данные передаются в и из SaaS-фирмы на скорости интернета, а не потенциально более высокие скорости внутренней сети фирмы.

#### Преимущества SaaS:

1. Для предоставления приложения SaaS пользователям, нет необходимости приобретать, устанавливать, обновлять или обслуживать оборудование, ПО промежуточного слоя или программное обеспечение. Если у организаций нет ресурсов для приобретения, развертывания необходимой инфраструктуры и программного обеспечения таких сложных промышленных приложений, как ERP или CRM, а также для управления ими, то можно сделать их доступными через SaaS.
2. Оплачиваются только те ресурсы, которые используются. SaaS автоматически изменяет масштабирование в соответствии с уровнем использования.
3. Пользователи могут запускать большинство приложений SaaS напрямую через веб-браузер без необходимости загружать и устанавливать программное обеспечение, хотя для некоторых приложений требуются подключаемые модули. Это означает, что нет необходимости приобретать и устанавливать специальное программное обеспечение для пользователей.
4. SaaS повышает мобильность сотрудников, так как пользователи имеют доступ к приложениям и данным SaaS с любого компьютера или мобильного устройства, которые подключены к Интернету. Кроме того, нет необходимости нанимать квалифицированных специалистов для решения проблем с безопасностью, которые могут возникнуть при использовании мобильных устройств.
5. Доступ к данным приложений из любого места. Когда данные хранятся в облаке, пользователи получают доступ к нужной информации с любого компьютера или мобильного устройства, подключенных к Интернету. Данные не могут быть потеряны в результате сбоя компьютера или устройства.

Чаще других, в качестве примеров использования SaaS-решений, можно встретить системы управления проектами, и совместной работы над ними, онлайн-органайзеры, системы документооборота. Для работы с документами — популярный google docs, который позволяет отказаться от word, excel, и получить ряд преимуществ, в первую очередь связанных с возможностями совместной работы над документами. Такие решения есть у целого ряда компаний — есть и у microsoft, и у компании Zoho и прочих. Существует достаточно много онлайн-органайзеров, как и вообще систем для организации работы, ведения todo. Не очевидный пример — онлайн-игры. Их не принято относить к SaaS-решениям, но и они на сегодня стали доступны как сервисы, со все той же идеологией. Чаще всего это MMORPG, к примеру, Xbox Live Microsoft очень четко себя классифицирует как SaaS-продукт.

Крупные поставщики услуг SaaS:

Google (Google Docs, Gmail).

Microsoft (Windows Azure).

Amazon (Amazon Web Services).

Все они обладают мощными инструментами для создания и поддержания работоспособности и доступности приложений под любые нужды клиента.

Сравнение

Расширение интернета в 1990-х годах привело к новому классу централизованных вычислений, называемому Application Service Providers. ASP предоставляют компаниям услуги по хостингу и управлению специализированными бизнес-приложениями с целью снижения затрат за счет централизованного администрирования и специализации поставщика решения в конкретном бизнес-приложении. SaaS существенно расширяет представление о модели ASP. Термин "программное обеспечение как услуга" подразумевает использование его в более конкретном случае. В то время как большинство ASP-программ были ориентированы на управление и поддержку стороннего программного обеспечения независимых поставщиков программного обеспечения, с 2012 года поставщики SaaS обычно разрабатывают и управляют собственными программными продуктами. В то время как многие первоначальные ASP предлагали более традиционные клиент-серверные приложения, требующие установки дополнительного ПО на персональные компьютеры пользователей,

сегодня решения SaaS опираются преимущественно на Web и требуют только веб-браузер для использования. В то время как в архитектуре программного обеспечения, используемой большинством ASP, требуется поддерживать определенный экземпляр приложения для каждой компании, решения SaaS используют многопользовательскую архитектуру, в которой приложение обслуживает несколько предприятий и пользователей и соответственно распределяет данные.

Модель ASP часто сравнивается с программным обеспечением как службой (SaaS), но в то время, как последняя обычно предоставляет универсальную услугу для многих пользователей, первая обычно включала предоставление услуги небольшому числу пользователей (часто с использованием отдельного для арендатора экземпляра). Это означало, что многие преимущества многопользовательской аренды (разделение затрат, экономия на масштабе и т. д.) не были доступны провайдерам ASP, а их услуги были более сопоставимы с внутренним хостингом, чем с настоящими многопользовательскими решениями SaaS, такими как Salesforce.com.

## Выводы

Исторически модель ASP, получившую распространение в 90-х гг. 20 в., можно считать своего рода предшественницей получившей сегодня массовое распространение модели SaaS. Провайдеры ASP-сервисов устанавливали программное обеспечение на своих серверных мощностях и предоставляли конечным пользователям доступ к этому программному обеспечению через Интернет. Развитие технологий виртуализации позволило вывести подобные сервисы на качественно новый уровень. В частности, автоматизировать процесс установки ПО и динамически выделять необходимые вычислительные ресурсы. Не стремясь к терминологической точности, сегодняшние SaaS-сервисы отличаются большой гибкостью в распределении ресурсов и масштабируемостью, что в конечном счете снижает скорость получения услуги и стоимость сервиса для конечного пользователя.