

ВИРТУАЛИЗАЦИЯ, ОБЛАЧНЫЕ ВЫЧИСЛЕНИЯ

ВИРТУАЛИЗАЦИЯ

ПРИЧИНЫ ВИРТУАЛИЗАЦИИ

- ◎ средний уровень загрузки Windows серверов ~ 5%
- ◎ Unix-серверы ~ 10-20%
- ◎ Подход "одно приложение — один сервер":
 - ◎ быстрое увеличение серверного парка
 - ◎ возрастание затрат на администрирование, энергопотребление и охлаждение
 - ◎ потребность в дополнительных помещениях

ВИРТУАЛИЗАЦИЯ

- ⊙ В основе виртуализации лежит возможность одного компьютера выполнять работу нескольких компьютеров благодаря распределению его ресурсов по нескольким средам.
- ⊙ Применение виртуализации позволяет распределять вычислительные ресурсы между приложениями, каждое из которых при этом "видит" только предназначенные ему ресурсы и "считает", что ему выделен отдельный сервер.
- ⊙ Кроме того, решения виртуализации дают возможность запускать в разделах разные ОС с помощью эмуляции их системных вызовов к аппаратным ресурсам сервера

ИСТОРИЯ ВИРТУАЛИЗАЦИИ

- ⊙ Компания IBM была первой, кто задумался о создании виртуальных сред для различных пользовательских задач, тогда еще в мэйнфреймах (1967 г. – платформа IBM CP-40)
- ⊙ После появления персональных компьютеров интерес к виртуализации несколько ослаб ввиду бурного развития операционных систем, которые предъявляли адекватные требования к аппаратному обеспечению того времени.
- ⊙ Однако бурный рост аппаратных мощностей компьютеров в конце девяностых годов прошлого века заставил ИТ-сообщество вновь вспомнить о технологиях виртуализации программных платформ.

ИСТОРИЯ ВИРТУАЛИЗАЦИИ

- ◎ В 1999 г. компания VMware представила технологию виртуализации систем на базе x86.
- ◎ Позднее в "битву" за место в этом модном направлении развития информационных технологий включились такие компании как
 - ◎ Parallels (ранее SWsoft),
 - ◎ Oracle (Sun Microsystems),
 - ◎ Citrix Systems (XenSource),
 - ◎ Microsoft (в 2003 г., купив Connectix и выпустив свой первый продукт Virtual PC).

ПРЕИМУЩЕСТВА ВИРТУАЛИЗАЦИИ

- ◎ Эффективное использование вычислительных ресурсов.
- ◎ Сокращение расходов на инфраструктуру.
- ◎ Снижение затрат на программное обеспечение.
- ◎ Повышение гибкости и скорости реагирования системы.
- ◎ Несовместимые приложения могут работать на одном компьютере.
- ◎ Повышение доступности приложений и обеспечение непрерывности работы предприятия.
- ◎ Возможности легкой архивации.
- ◎ Повышение управляемости инфраструктуры.

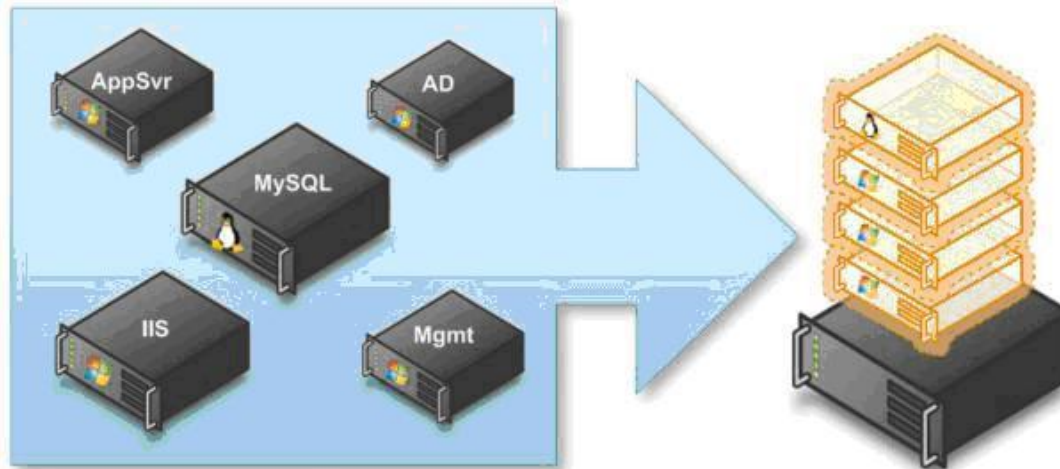
ВИРТУАЛЬНАЯ МАШИНА

- ◎ **Виртуальной машиной** будем называть программную или аппаратную среду, которая скрывает настоящую реализацию какого-либо процесса или объекта от его видимого представления.
- ◎ **Виртуальная машина** — это полностью изолированный программный контейнер, который работает с собственной ОС и приложениями, подобно физическому компьютеру. Виртуальная машина действует так же, как физический компьютер, и содержит собственные виртуальные (т.е. программные) ОЗУ, жесткий диск и сетевой адаптер.



ВИРТУАЛИЗАЦИЯ СЕРВЕРОВ

- ◎ Виртуализация серверов подразумевает запуск на одном физическом сервере нескольких виртуальных серверов.
- ◎ На каждой виртуальной машине может быть установлена операционная система, на которую могут быть установлены приложения и службы.
- ◎ Типичные представители это продукты VmWare (ESX, Server, Workstation) и Microsoft (Hyper-V, Virtual Server, Virtual PC).



ГИПЕРВИЗОР

- ☉ **Монитор Виртуальных Машин (Гипервизор, Virtual Machine Monitor, VMM)** берет на себя управление гостевыми системами. Его можно рассматривать как абстракцию между аппаратной платформой и виртуальными машинами.
- ☉ В некоторых случаях гипервизор является операционной системой; в этом случае он называется базовой операционной системой.



Виды виртуализации

Эмуляция оборудования

- ① Гостевая ОС запускается на виртуальной машине, которая перехватывает каждую команду и моделирует ее на реальном аппаратном обеспечении.
- ① Нет привязки к архитектуре host-машины (можно запускать неизмененные операционные системы, предназначенные для машин с другой архитектурой).
- ① Очень низкая скорость работы.

ПОЛНАЯ ВИРТУАЛИЗАЦИЯ

- ⊙ Используются не модифицированные экземпляры гостевых операционных систем, а для поддержки работы этих ОС служит общий слой эмуляции их исполнения поверх хостовой ОС
- ⊙ Гипервизор перехватывает только некоторые опасные команды, все остальные напрямую выполняются на аппаратном обеспечении.
- ⊙ VMware Workstation, VMware Server (бывший GSX Server), Parallels Desktop, Parallels Server, MS Virtual PC, MS Virtual Server, Virtual Iron.



ПАРАВИРТУАЛИЗАЦИЯ

- ⊙ Модификация ядра гостевой ОС выполняется таким образом, что в нее включается новый набор API, через который она может напрямую работать с аппаратурой, не конфликтуя с другими виртуальными машинами.
- ⊙ VMware ESX Server, Xen (и решениях других поставщиков на базе этой технологии), Microsoft Hyper-V.



ВИРТУАЛИЗАЦИЯ НА УРОВНЕ ЯДРА ОС

- ⊙ Использование одного ядра хостовой ОС для создания независимых параллельно работающих операционных сред. Для гостевого ПО создается только собственное сетевое и аппаратное окружение.
- ⊙ Virtuozzo (для Linux и Windows), OpenVZ (бесплатный вариант Virtuozzo) и Solaris Containers.



ВИРТУАЛИЗАЦИЯ ПРИЛОЖЕНИЙ

- ◎ Сильная изоляция прикладных программ с управляемым взаимодействием с ОС, при которой виртуализируется каждый экземпляр приложений, все его основные компоненты: файлы (включая системные), реестр, шрифты, INI-файлы, COM-объекты, службы.
- ◎ Sun Java Virtual Machine, Microsoft Application Virtualization (ранее называлось Softgrid), Thinstall (в начале 2008 г. вошла в состав VMware), Symantec/Altiris



ПЛАТФОРМЫ ВИРТУАЛИЗАЦИИ

- ◎ Компания VMware – один из первых игроков на рынке платформ виртуализации.
- ◎ В 1998 году VMware запатентовала свои программные техники виртуализации и с тех пор выпустила немало эффективных и профессиональных продуктов для виртуализации различного уровня:
 - ◎ VMware Workstation, предназначен для настольных ПК
 - ◎ VMware ESX Server, позволяет консолидировать физические серверы предприятия в виртуальной инфраструктуре.

CITRIX (XEN)

- ⊙ Некоммерческий гипервизор - Xen исследовательский проект компьютерной лаборатории Кембриджского университета.
- ⊙ Технология позволяет гипервизору в хостовой системе управлять гостевой ОС посредством гипервызовов VMI (Virtual Machine Interface), что требует модификации ядра гостевой системы.
- ⊙ Бесплатная версия Xen включена в дистрибутивы нескольких ОС, таких как Red Hat, Novell SUSE, Debian, Fedora Core, Sun Solaris

- ◎ Microsoft в 2003 году она приобрела компанию Connectix (система Virtual PC). Virtual PC по своим возможностям сопоставим с VMware Player
- ◎ 2006 год: Microsoft Virtual Server 2005 – позволяет обеспечить запуск нескольких виртуальных машин на сервере.
- ◎ 2008 год: Microsoft Hyper-V обеспечивает виртуализацию на аппаратном уровне, с использованием технологий виртуализации, встроенных в процессоры.

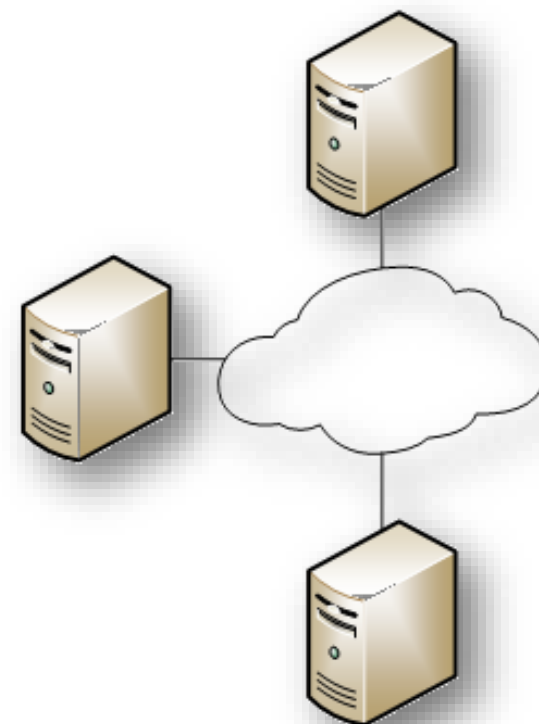
ОБЛАКА

Откуда пошли «Облака» ?

Символ «облако» традиционно использовался для обозначения сложной вычислительной инфраструктуры, внутреннее устройство которой не важно с точки зрения конечного пользователя.

«Модным» этот термин стал в конце 2007-начале 2008 года с подачи компании IBM.

В коммерческом мире он заменил термин «Грид».



КОММУНАЛЬНЫЕ ВЫЧИСЛЕНИЯ

- ◎ Предвестник облачных вычислений – «utility computing» (коммунальные вычисления) - предоставление данных и процессорных мощностей организовано по принципам коммунальных услуг, по принципу «оплата по мере использования».
- Feeney G.J., Hilton R.D., Johnson R.L., O'Rourke T.J., Kurtz T.E. Utility computing: a superior alternative? // Proceedings of the AFIPS '74 national computer conference and exposition. ACM, 1974. P. 1003-1004.

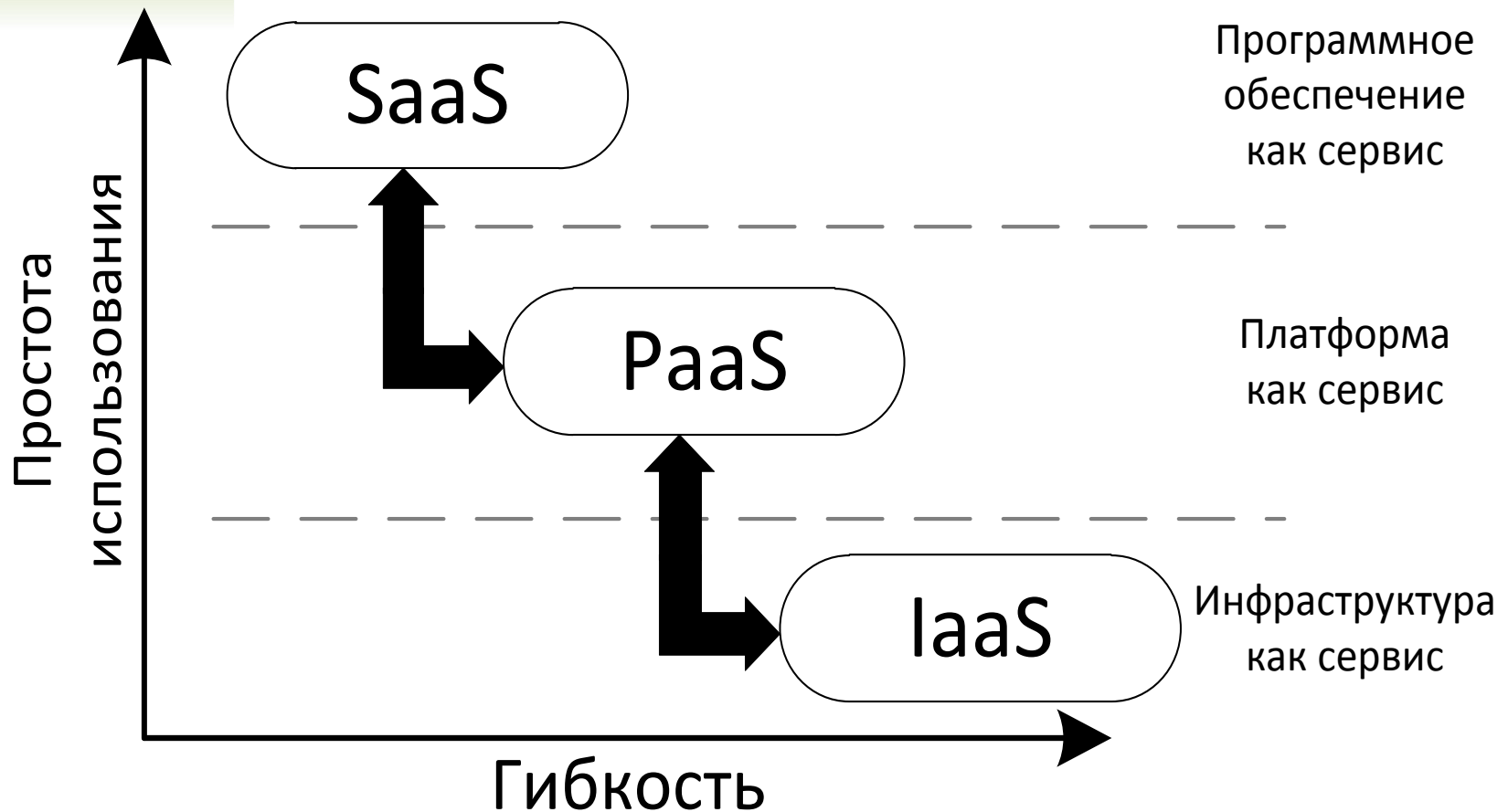
ОБЛАЧНЫЕ ВЫЧИСЛЕНИЯ

- ◎ Облачные вычисления – это модель предоставления по требованию повсеместного, удобного сетевого доступа к общему пулу абстрактных, виртуализованных, динамически-масштабируемых вычислительных ресурсов (таких как сети, серверы, хранилища, приложения, сервисы).
- ◎ Эти ресурсы динамически масштабируются для подстройки под текущую нагрузку, обеспечивая оптимальное использование и обычно предоставляется по принципу «оплата по мере использования».

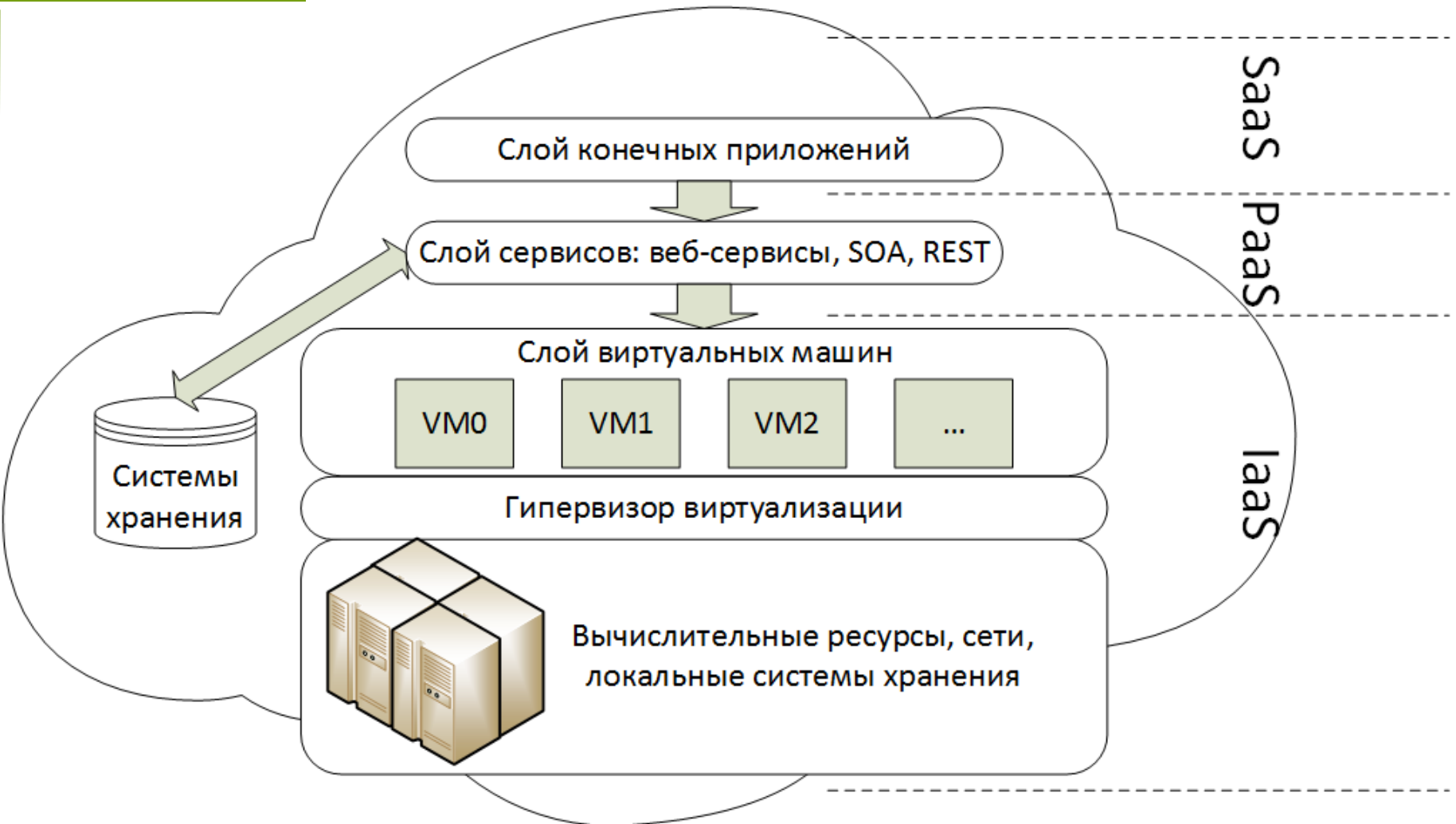
ХАРАКТЕРИСТИКИ ОБЛАЧНЫХ СИСТЕМ

- ◎ *Самообслуживание по требованию.* Пользователь может самостоятельно выделять себе требуемый объем ресурсов.
- ◎ *Повсеместный сетевой доступ.* Ресурсы облачных платформ доступны по сети посредством стандартных сетевых механизмов.
- ◎ *Объединение ресурсов.* Вычислительные ресурсы объединяются для обслуживания множества различных потребителей.
- ◎ *Быстрая эластичность.* Вычислительные возможности могут быть эластично предоставлены и перераспределены.
- ◎ *Измеряемый объем предоставленных услуг.* Облачные системы автоматически контролируют и оптимизируют использование ресурсов

КЛАССИФИКАЦИЯ ОБЛАКОВ



ОБЛАКА - ОНИ КАК ЛУК, СО СЛОЯМИ...



МОДЕЛИ ВНЕДРЕНИЯ

- ◎ **Частные облака:** могут быть развернуты в рамках частного ЦОД для оптимизации используемых ресурсов (оптимизация коэффициента эффективной загрузки с 5% до 15%)
- ◎ **Публичные облака:** центры обработки данных, предоставляющие свои ресурсы третьим лицам через Интернет (например, Google или Amazon).
- ◎ **Гибридные облака:** объединяют общественные и частные облака, позволяя организациям запускать часть приложений внутри частного облака, а другую часть данных передавать на обработку в общественное облако.
- ◎ **Федерации облаков:** пока-что утопия, или реализуются отдельно для каждого приложения.

ОБЛАЧНЫЕ ПЛАТФОРМЫ

Платформы Характеристики	Amazon Web Services	Google App Engine	Microsoft Windows Azure
Тип	IaaS	PaaS	PaaS
Разрабатываемые сервисы	Вычислительные сервисы, сервисы хранения	Web-приложения	Как Web-приложения, так и не Web-приложения
Виртуализация	Уровня ОС, с запущенным гипервизором Xen	Контейнер приложений	Уровня ОС
Интерфейс доступа пользователя	Утилиты консоли Amazon EC2	Web-консоль администрирования	Портал Microsoft Windows Azure
Web APIs	Да	Да	Да
Среда разработки	Отсутствует	Python, Java	Microsoft .NET

ДОСТОИНСТВА ОБЛАКОВ

- ⊙ Отсутствует необходимость предварительных инвестиций в инфраструктуру
- ⊙ Покупатели облачных сервисов покупают и используют только тот объем информационных ресурсов, который им на самом деле необходим
- ⊙ Высокая масштабируемость и гибкость
- ⊙ Хорошая документация, развитые API, и наборы готовых сервисов
- ⊙ Высокая географическая распределенность и близость к конечному пользователю

НЕДОСТАТКИ ОБЛАКОВ

- ⊙ Отсутствие единого универсального «облачного интерфейса», каждое приложение приходится затачивать под определенную облачную платформу
- ⊙ Облачные системы могут прекратить поставку облачных решений и закрыть используемые сервисы
- ⊙ Невозможно контролировать собственное окружение, неизвестно, с кем рядом сидит твоя виртуальная машина
- ⊙ Высокие риски безопасности: передача данных через открытую сеть (в/из облака); обработка данных на удаленных, не подконтрольных системах