

Распределенные системы

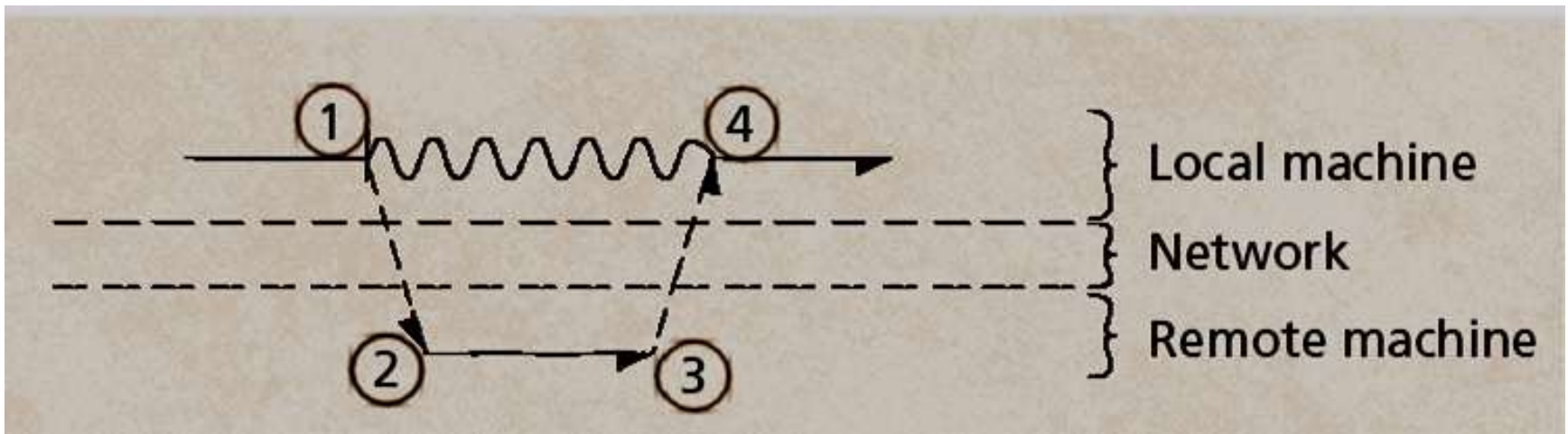
Связь в распределенных
системах

Удаленный вызов процедур

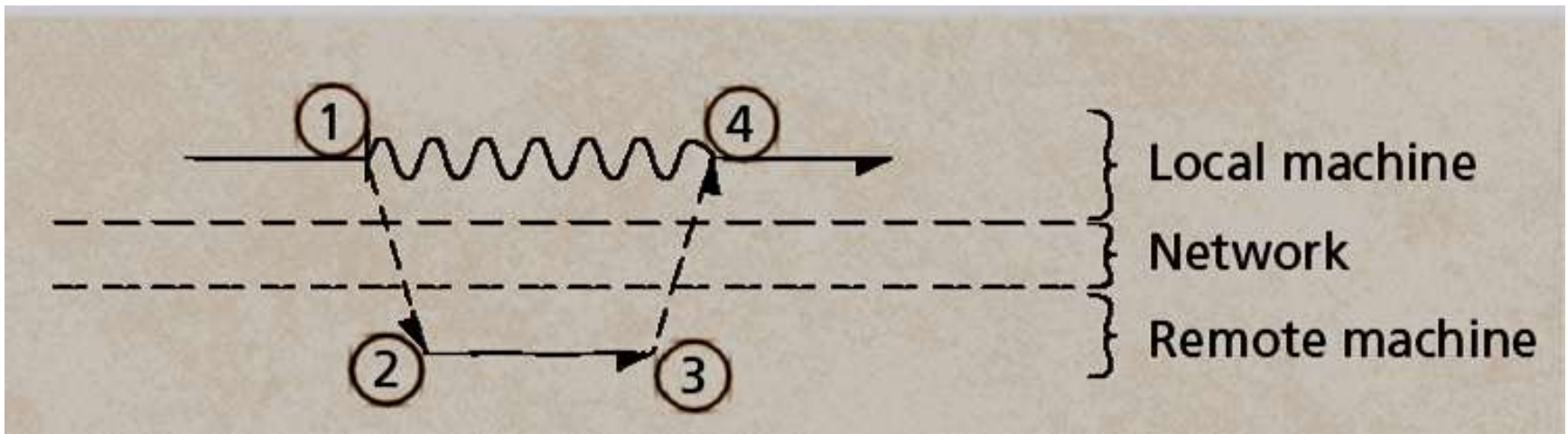
Опр. Удаленный вызов процедур (Remote Procedure Call, RPC) – позволяет процессу, выполняющемуся на одном компьютере (клиенте), вызывать процедуры процесса, выполняющегося на другом компьютере (сервере). В RPC используется концепция заглушек.

Заглушка

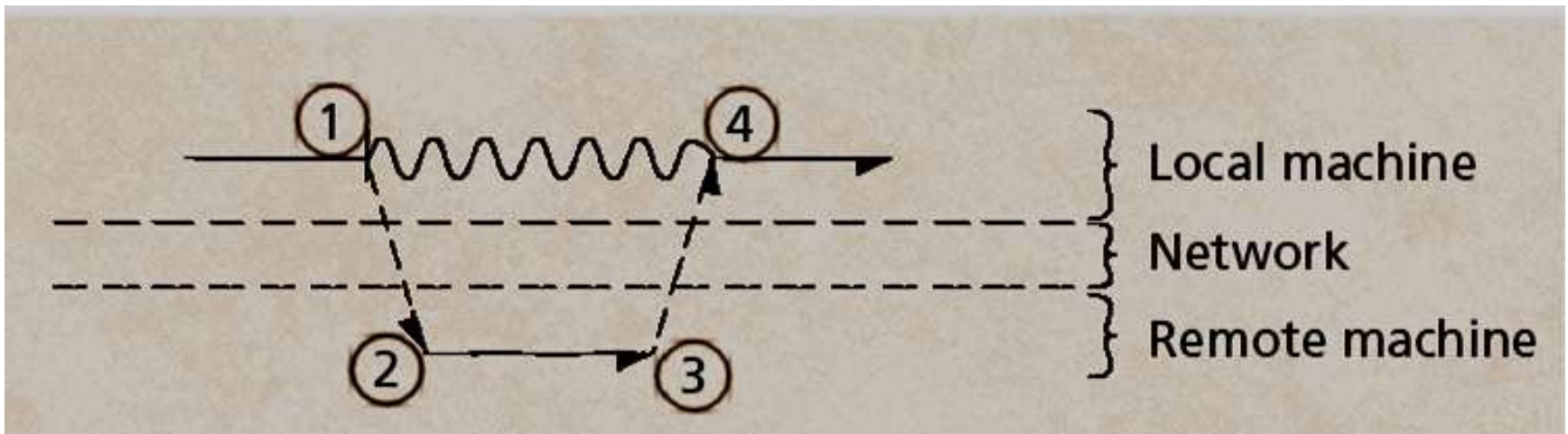
Опр. Заглушка (stub) – специальный процесс, который готовит исходящие данные к передаче и преобразует входящие данные, чтобы те могли корректно интерпретироваться. Заглушки располагаются и на клиенте и на сервере.



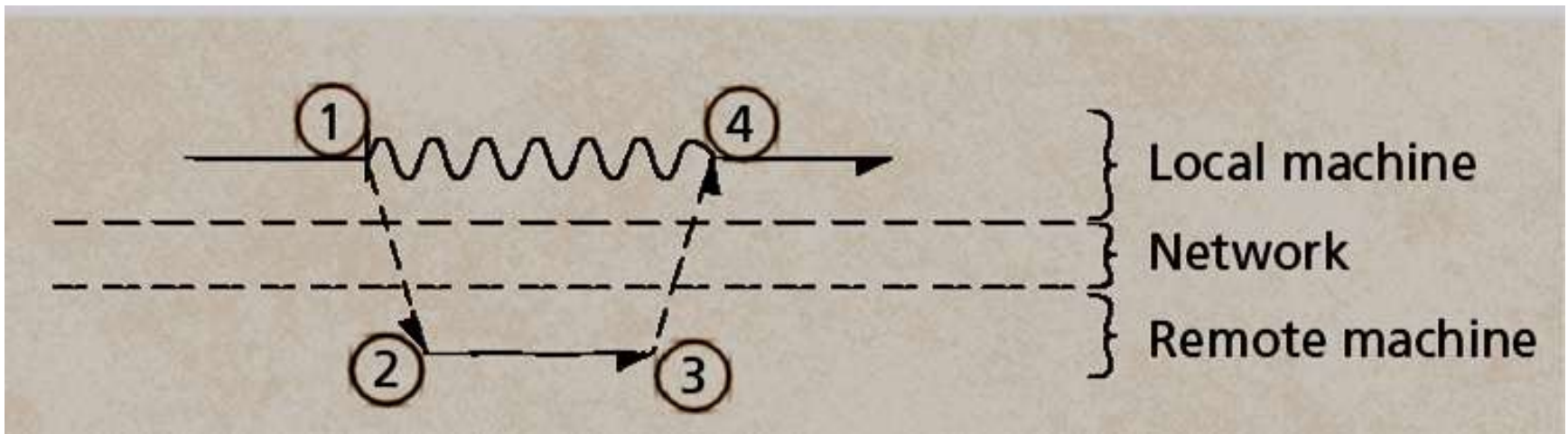
1. Локальный процесс вызывает процедуру через клиентскую заглушку. Процесс блокируется, заглушка упаковывает параметры и передает запрос на сервер.



2. Серверная заглушка принимает запрос, распаковывает параметры и передает параметры соответствующему процессу на сервере.



3. Серверный процесс завершает работу вызванной процедуры, заглушка на сервере упаковывает результаты и возвращает их клиенту.



4. Клиентская заглушка распаковывает полученные результаты, уведомляет об их получении процесс, отправивший запрос и передает данные этому процессу.

Вопрос для самопроверки

- Механизм RPC позволяет вызывать процедуры на сервере так же легко, как на локальной системе? (Да/Нет)

Вопрос для самопроверки

- Механизм RPC позволяет вызывать процедуры на сервере так же легко, как на локальной системе? (Да/Нет)
- Да. Цель RPC – упрощение процесса написания распределенных приложений, при условии сохранения синтаксиса, используемого для вызова локальных процедур.

Вопрос для самопроверки

- Поддерживает ли RPC глобальные переменные? (Да/Нет)

Вопрос для самопроверки

- Поддерживает ли RPC глобальные переменные? (Да/Нет)
- Нет. Процесс, осуществляющий удаленный вызов процедуры и соответствующая ему клиентская заглушка имеют различные адресные пространства данных и не могут иметь общие глобальные переменные.

Удаленный вызов методов

Опр. Удаленный вызов методов (Remote Method Invocation, RMI) является аналогом RPC в языке Java и позволяет сценариям Java выполнять обращение к методам объекта на другом компьютере, используя тот же самый синтаксис, что и при вызове локальных методов.

RMI

- Заглушки - объекты Java, обеспечивающие интерфейс между клиентскими процессами и удаленными объектами
- Объекты передаются между удаленными процессами в упакованном виде в качестве параметров

Вопрос для самопроверки

- Обладает ли преимуществом механизм RMI перед RPC? (Да/Нет)

Вопрос для самопроверки

- Обладает ли преимуществом механизм RMI перед RPC? (Да/Нет)
- Да. RMI позволяет клиентам отправлять объекты в качестве аргументов при удаленном вызове методов и принимать объекты в качестве возвращаемых значений от серверов.

CORBA

Опр. Типовая архитектура брокера объектных запросов (Common Object Request Broker Architecture, CORBA) – стандартная технология построения распределенных объектных приложений, принятая в начале 90-х рабочей группой по развитию стандартов объектного программирования (Object Management Group, OMG).

DCOM

Опр. Распределенная объектная модель программных компонентов (Distributed Component Object Model, DCOM) – архитектура распределенных объектов, разработанная независимо от других разработчиков в начале 90-х компанией Microsoft. Встроена в операционную систему Windows начиная с Windows 95.

CORBA и DCOM

- Подобно RMI поддерживают передачу объектов между процессами
- Независимы от языков программирования
- Приложения, написанные на разных языках, могут взаимодействовать через интерфейсы CORBA или DCOM

Вопрос для самопроверки

- Может ли пользователь отдать предпочтение RMI вместо CORBA?
(Да/Нет)

Вопрос для самопроверки

- Может ли пользователь отдать предпочтение RMI вместо CORBA? (Да/Нет)
- Да. Если пользователь имеет дело только с приложениями, написанными на языке Java, он ничего не выигрывает от языковой независимости CORBA, но должен дополнительно изучить интерфейсы CORBA.

Вопрос для самопроверки

- Есть ли общее между DCOM и CORBA?
(Да/нет)

Вопрос для самопроверки

- Есть ли общее между DCOM и CORBA?
(Да/нет)
- Да. В обе эти технологии включена поддержка удаленных объектов, написанных для разных платформ с использованием различных языков программирования.

Распределенные системы

Веб-службы

Веб-служба

Опр. Веб-служба, веб-сервис (web service) — программная система, чьи общедоступные интерфейсы определены на языке XML. Описание этой программной системы может быть найдено другими программными системами, которые могут взаимодействовать с ней согласно этому описанию посредством сообщений, основанных на XML, и передаваемых с помощью интернет-протоколов.

XML

Опр. Расширяемый язык разметки (eXtensible Markup Language, XML) — рекомендованный Консорциумом Всемирной паутины язык разметки, представляющий собой свод общих синтаксических правил. XML это текстовый формат, предназначенный для хранения структурированных данных, для обмена информацией между программами, а также для создания на его основе более специализированных языков разметки.

Веб-службы

- Обеспечивают взаимодействие программных систем независимо от платформы
- Основаны на базе открытых стандартов и протоколов. Благодаря использованию XML достигается простота разработки и отладки веб-служб
- Меньшая производительность и больший размер сетевого трафика по сравнению с технологиями RMI, CORBA, DCOM за счёт использования текстовых XML-сообщений

.NET Framework

Опр. .NET Framework — программная платформа, выпущенная компанией Microsoft в 2002 году. Фактически представляет собой операционную систему внутри операционной системы. Основой платформы является виртуальная машина Common Language Runtime (CLR), способная выполнять как обычные настольные программы, так и веб-приложения. Отличительной особенностью .NET Framework является способность выполнять программы, написанные на разных языках программирования.

Архитектура .NET

- Программа для .NET Framework, написанная на любом поддерживаемом языке программирования, сначала переводится компилятором в единый для .NET понятный человеку низкоуровневый язык Common Intermediate Language (CIL)
- Затем компилятор производит перевод CIL-кода в объектный байт-код (в терминах .NET получается сборка, assembly), а уже байт-код исполняется виртуальной машиной CLR

Архитектура .NET

- Встроенный в виртуальную машину JIT-компилятор «на лету» (just in time — компиляция на лету) преобразует промежуточный байт-код в машинные коды нужного процессора
- Современная технология динамической компиляции позволяет достигнуть высокого уровня быстродействия
- Виртуальная машина CLR также сама заботится о базовой безопасности и управлении памятью, избавляя разработчика от части работы

Программная совместимость

- Одной из основных идей Microsoft .NET является совместимость программных частей, написанных на разных языках
- Служба, написанная на C++ для Microsoft .NET, может обратиться к методу класса из библиотеки, написанной на Delphi
- На C# можно написать класс, наследованный от класса, написанного на Visual Basic .NET
- Исключение, созданное методом, написанным на C#, может быть перехвачено и обработано в Delphi

Вопрос для самопроверки

- Может ли пользователь отдать предпочтение DCOM перед .NET?
(Да/Нет)

Вопрос для самопроверки

- Может ли пользователь отдать предпочтение DCOM перед .NET?
(Да/Нет)
- Да. Если важна высокая производительность и ограничение сетевого трафика.

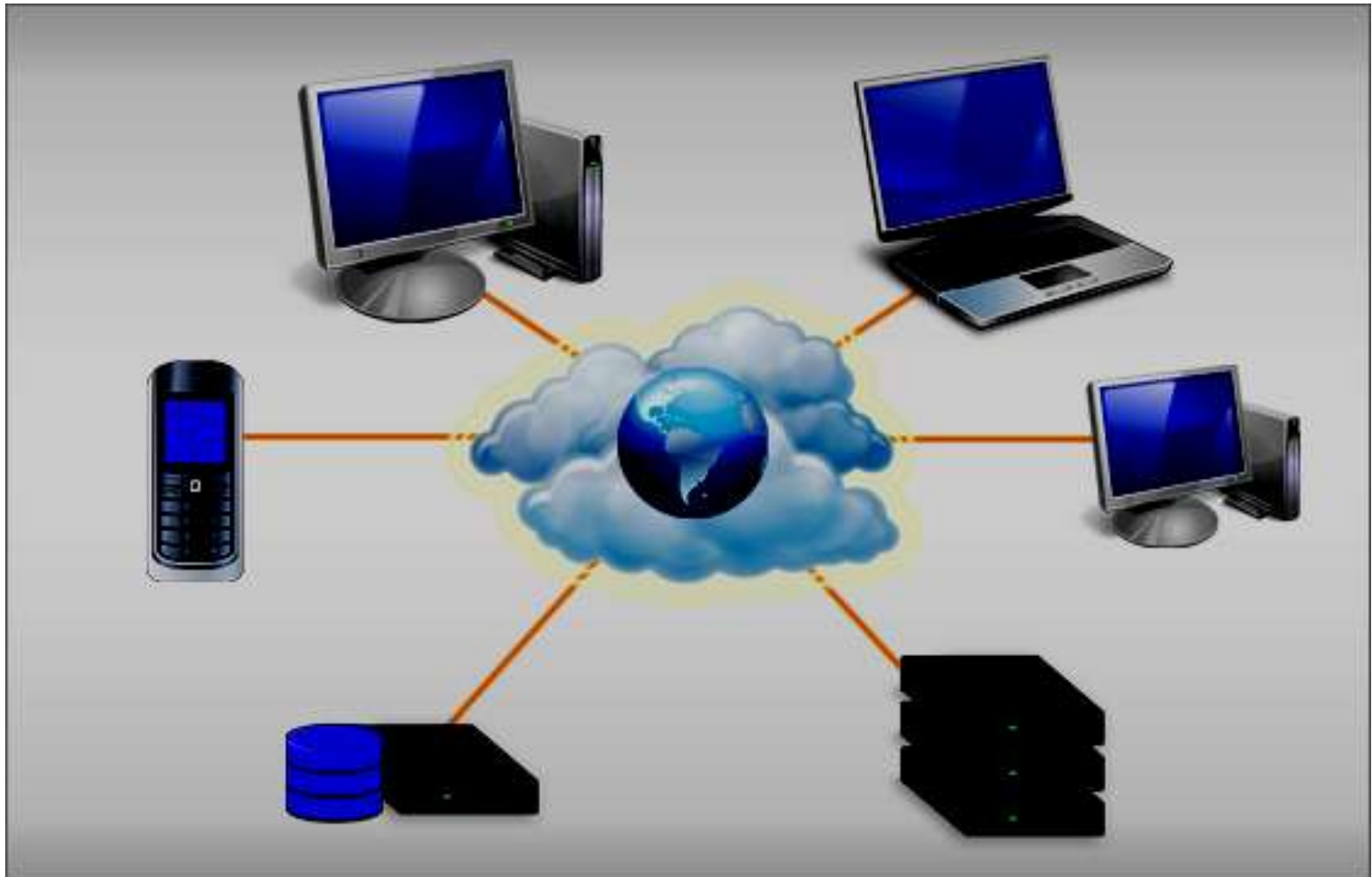
Распределенные системы

Облачные вычисления

Облачные вычисления

Опр. Облачные вычисления (cloud computing) — технология распределённой обработки данных, в которой компьютерные ресурсы и мощности предоставляются пользователю как интернет-сервис.

Облачные вычисления



IEEE, 2008 год:

- Облачная обработка данных — это парадигма, в рамках которой информация постоянно хранится на серверах в Интернете и временно кэшируется на клиентской стороне, например, на персональных компьютерах, игровых приставках, ноутбуках, смартфонах и так далее.

Облачная обработка данных:

- Программное обеспечение как услуга
- Инфраструктура как услуга
- Все как услуга

Программное обеспечение как услуга

Опр. Программное обеспечение как услуга (software as a service, SaaS) — бизнес-модель продажи и использования программного обеспечения, при которой поставщик разрабатывает веб-приложение и самостоятельно управляет им, предоставляя заказчикам доступ к программному обеспечению через Интернет. Основное преимущество модели SaaS для потребителя состоит в отсутствии затрат, связанных с установкой, обновлением и поддержкой работоспособности оборудования и работающего на нём программного обеспечения.

SaaS

- Приложение приспособлено для удаленного использования
- Одним приложением пользуется несколько клиентов (приложение коммунально)
- Оплата взимается либо в виде ежемесячной абонентской платы, либо на основе объема операций
- Техническая поддержка приложения включена в оплату
- Модернизация и обновление приложения происходит плавно и прозрачно для клиентов

SaaS

- Заказчики платят не за владение программным обеспечением как таковым, а за его аренду (то есть за его использование через веб-интерфейс)
- Заказчик несет сравнительно небольшие периодические затраты, и ему не требуется инвестировать значительные средства в приобретение ПО и аппаратной платформы для его развертывания, 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

Вопрос для самопроверки

- Позволяет ли SaaS бороться с нелегальным ПО? (Да/Нет)

Вопрос для самопроверки

- Позволяет ли SaaS бороться с нелегальным ПО? (Да/Нет)
- Да. Модель SaaS позволяет эффективно бороться с нелегальным использованием программного обеспечения, поскольку само программное обеспечение не попадает к конечным заказчикам.

Инфраструктура как услуга

Опр. Инфраструктура как услуга (Infrastructure as a Service, IaaS) — это предоставление компьютерной инфраструктуры (как правило в форме виртуализации) как услуги на основе концепции облачных вычислений.

Компоненты IaaS

- Аппаратные средства (серверы, системы хранения данных, клиентские системы, сетевое оборудование)
- Операционные системы и системное ПО (средства виртуализации, автоматизации, основные средства управления ресурсами)
- Связующее ПО (например, для управления системами)

IaaS

- Избавляет предприятия от необходимости поддержки сложных инфраструктур центров обработки данных, клиентских и сетевых инфраструктур
- Позволяет уменьшить связанные с этим капитальные затраты и текущие расходы
- Возможна и дополнительная экономия, если услуги предоставляются в рамках инфраструктуры совместного использования

Вопрос для самопроверки

- Может ли IaaS приводить к возникновению неконтролируемых данных? (Да/Нет)

Вопрос для самопроверки

- Может ли IaaS приводить к возникновению неконтролируемых данных? (Да/Нет)
- Да. Информация, оставленная пользователем, может храниться годами, либо использоваться без его ведома, либо он будет не в состоянии изменить какую-то её часть.