**Некоммерческое акционерное общество**
**«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ»**

Кафедра Информационные системы
Специальность М ИСп 12

Допущен к защите
Зав. кафедрой ИС

_____
«____»_____2014 г.

# МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ
## пояснительная записка

Тема Автоматизация и оптимизация учебного процесса в университетах РК, на основе инновационных технологий

Магистрант Нурлыбаев Р.Е
подпись (Ф.И.О.)

Руководитель диссертации Нурлыбаев Н.А
подпись (Ф.И.О.)

Рецензент _____
подпись (Ф.И.О.)

Алматы, 2014 г.

**Некоммерческое акционерное общество**
**«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ»**

Факультет ФИТ
Специальность Информационные системы и ПО
Кафедра Информационные системы

## ЗАДАНИЕ

на выполнение магистерской диссертации

Магистранту Нуртбекову Райымбеку Ерсултановичу
(фамилия, имя, отчество)

Тема диссертации Автоматизация и оптимизация учебного процесса в университетах на основе интеллектуальных технологий
утверждена Ученым советом университета №___ от «___»_____
Срок сдачи законченной диссертации «___»_____
Цель исследования

Перечень подлежащих разработке в магистерской диссертации вопросов или краткое содержание магистерской диссертации:
Исследование вопроса внутривузовских тонких каналов и внутривузовских приложений

Перечень графического материала (с точным указанием обязательных чертежей)
График роста внутривузовских схема подключения тонких каналов

## Аннотация

Рекомендуемая основная литература

Г Р А Ф И К
подготовки магистерской диссертации

| Наименование разделов, перечень разрабатываемых вопросов | Сроки представления научному руководителю | Примечание |
|---|---|---|
| 1. Виртуалды зертхана | 01.02.14 | |
| 2. Центр образования данных | 01.03.14 | |
| 3. Разработка виртуальной библиотеки | 01.04.14 | |

Дата выдачи задания_____

Заведующий кафедрой Шаншалиев Ш.А. (_____)
(подпись) (Ф.И.О.)

Руководитель диссертации Нурмыбаев Н.Ф. (_____)
(подпись) (Ф.И.О.)

Задание принял к исполнению
магистрант Нурмуханов Р.Е (_____)
(подпись) (Ф.И.О.)

Біздің заманда ақпараттық технологияларға қажет «темір» құрылғылардың тез ескіріп оны әрдайым жаңартудың қажеттілігін көреміз. Бұл өте қымбат бағалы процесстер. Бұлтты технологиялар бұл мәселелерді шешуге көмек етеді, бірақ бар процесстерді басқаратын функциялары болатын программалық жабдықтарды құрастыруды қажет етеді. Оқыту

процесстерін автоматизациялау ісінің негізінде қажетті оптимизация жасалып инновациялық виртуализация технологиялары мен бұлтты әдістемелер қолданылуы көрсетілді. Осы мақалада жіңішке клиенттерді құрастырып еңгізу және оқыту қолданбаларды виртуализациялау амалдары мен мәселелері қарастырылды.

## Аннотация

В наше время мы наблюдаем проблему быстрого устаревания аппаратного “железа” и постоянную необходимость его обновления. Это очень дорогостоящие процессы. Облачные технологии позволяют решить эту проблему, но, требуют создания программного обеспечения, который возьмет на себя функцию управления всеми процессами. На примере задачи автоматизации учебного процесса была проведена его оптимизация и реализация на основе инновационных технологий виртуализации и облачных вычислений. В данной статье рассматриваются вопросы внедрения тонких клиентов и виртуализации обучающих приложений.

## Annotation

In our time we can see the problem of rapid obsolescence of hardware "hardware" and the constant need to update it .This is a very costly process. Cloud technology can solve this problem, but require the creation of software that will take over the function of the management of all processes. On an example of the problem of automation of the educational process was conducted its optimization and implementation of innovative technologies for virtualization and cloud computing. This article discusses the implementation of thin clients and virtualization of learning applications.

**Содержание**

Введение ..... 6
1 Виртуализация ..... 8
1.1 Описание виртуализации ..... 8
1.2 Виртуализация сегодня ..... 22
1.3 Преимущества и недостатки использования виртуальных машин ..... 23
2 Центр обработки данных ..... 40
2.1 Концепция ..... 40
2.2 Основные преимущества концепции ..... 43
2.3 Защита данных ..... 46
2.4 Экономическая сторона ..... 46
2.5 Окупаемость ..... 49
3 Разработка виртуальной библиотеки ..... 51
3.1 Организационная структура ..... 55
3.2 Перечень и описание основных сущностей предметной области ..... 55
3.3 Концептуальное проектирование ..... 56
Заключение ..... 62
Список литературы ..... 63
Определение ..... 65

## Введение

Первые электронные компьютеры появились в развитых странах на рубеже пятидесятых годов двадцатого века, и предназначались, как правило, для решения задач военного значения. Создатели компьютеров были также и первыми программистами и отладчиками, как правило были изделиями штучного производства. Естественно компьютеры производства разных стран были несовместимы программно и технологически.

В конце пятидесятых годов компания IBM выпустила впервые серийные универсальные компьютеры, которые могли использоваться для решения научных задач и бизнес процессов.

До середины шестидесятых годов крупнейшим производителем компьютеров была. В 1964 г. была компания General Electric**.**

В 1964 General Electric совместно с MIT и Bell Labs создали проект Multics**.**

Но у проекта было много противников и проект провалился, было продано всего 76 лицензий. В результате General Electric вышла из компьютерного бизнеса. Однако в проекте было много новых идей, которые воплотились позже. В частности в операционную систему Unix.

После эпохи Unix

Началась эпоха персональных компьютеров и мощный взрыв прикладных программ, компьютерных игр, неизбежных компьютерных вирусов,троянов,червей.

Начиная с 90 годов двадцатого столетия основными трендами развития ИТ стали технологии виртуализация и облачные вычисления. Сам термин «виртуализация» в компьютерных технологиях появился в шестидесятых годах прошлого века вместе с термином «виртуальная машина», означающим продукт виртуализации программно-аппаратной платформы. В то время виртуализация была, скорее, интересной технической находкой, чем перспективной технологией. Разработки в сфере виртуализации в шестидесятых-семидесятых годах проводились в основном компанией IBM.

С появлением в компьютере IBM M44/44X экспериментальной системы пэйджинга, впервые был употреблен термин «виртуальная машина» (virtual machine), который заменил более ранний термин «псевдо машина» (pseudo machine). Затем в мэйнфреймах IBM серии System 360/370, можно было использовать виртуальные машины для сохранения предыдущих версий операционных систем. До конца девяностых годов никто кроме IBM так и не решался использовать эту оригинальную технологию всерьез. Однако в девяностых годах стали очевидны перспективы подхода виртуализации: с ростом аппаратных мощностей, как персональных компьютеров, так и серверных решений, представляется возможность использовать несколько виртуальных машин на одной физической платформе. В 1997 году компания Connectix выпускает первую

версию Virtual PC для платформы Macintosh, а в 1998 году VMware патентует свои технологии виртуализации. Компания Connectix впоследствии была куплена корпорацией Microsoft, а VMware корпорацией EMC, и на данный момент обе эти компании являются двумя основными потенциальными конкурентами на рынке технологий виртуализа-ции в будущем. Потенциальными - потому что сейчас VMware безоговорочный лидер на этом рынке, Со времени своего появления термины «виртуализация» и «виртуальная машина» приобрели множество различных значений и употреблялись в разных контекстах. В настоящее время множество компаний стремятся войти в майнстрим компьютерных технологий. Современное состояние в технологиях виртуализации возглавляется компаниями VM Ware, Citrix, Microsoft и Oracle.

# 1 Виртуализация

## 1.1 Описание виртуализации

В настоящее время Информационные технологии являются основной движущей силой технического прогресса.. Множество различных программных и аппаратных средств, технологий и сервисов позволяют нам ежедневно повышать удобство и скорость работы с информацией. Все сложнее и сложнее выделить из обрушивающегося на нас потока технологий действительно полезные и научиться применять их с максимальной пользой. Именно такими технологиями являются технологии виртуализации и облачных вычислений. В широком смысле, понятие виртуализации представляет собой сокрытие настоящей реализации какого-либо процесса или объекта от истинного его представления для того, кто им пользуется. Продуктом виртуализации является нечто удобное для использования, на самом деле, имеющее более сложную или совсем иную структуру, отличную от той, которая воспринимается при работе с объектом. Иными словами, происходит отделение представления от реализации чего-либо. В компьютерных технологиях под термином «виртуализация» обычно понимается абстракция вычислительных ресурсов и предоставление пользователю системы, которая «инкапсулирует» (скрывает в себе) собственную реализацию. Проще говоря, пользователь работает с удобным для себя представлением объекта, и для него не имеет значения, как объект устроен в действительности.

Сам термин «виртуализация» в компьютерных технологиях появился в шестидесятых годах прошлого века вместе с термином «виртуальная машина», означающим продукт виртуализации программно-аппаратной платформы. В то время виртуализация была, скорее, интересной технической находкой, чем перспективной технологией. Разработки в сфере виртуализации в шестидесятых-семидесятых годах проводились в основном компанией IBM. С появлением в компьютере IBM M44/44X экспериментальной системы пэйджинга, впервые был употреблен термин «виртуальная машина» (virtual machine), который заменил более ранний термин «псевдо машина» (pseudo machine). Затем в мэйнфреймах IBM серии System 360/370, можно было использовать виртуальные машины для сохранения предыдущих версий операционных систем. До конца девяностых годов никто кроме IBM так и не решался использовать эту оригинальную технологию всерьез. Однако в девяностых годах стали очевидны перспективы подхода виртуализации: с ростом аппаратных мощностей, как персональных компьютеров, так и серверных решений, вскоре представится возможность использовать несколько виртуальных машин на одной физической платформе.

В 1997 году компания Connectix выпускает первую версию Virtual PC для платформы Macintosh, а в 1998 году VMware патентует свои технологии

виртуализации. Компания Connectix впоследствии была куплена корпорацией Microsoft, а VMware корпорацией EMC, и на данный момент обе эти компании входят в большую четверку лидеров виртуализации. Со времени своего появления термины «виртуализация» и «виртуальная машина» приобрели множество различных значений и употреблялись в разных контекстах.

Так, например, в трудах ассоциации SNIA дается следующее общее определение: “Виртуализация — это действие (act) по объединению нескольких устройств, служб или функций внутренней составляющей инфраструктуры (back-end) с дополнительной внешней (front-end) функциональностью, обеспечивающее целесообразное абстрагирование от внутреннего устройства.” Обычно виртуализация позволяет скрыть от пользователя внутренние сложности и сделать его работу удобнее. Примерами виртуализации являются агрегация нескольких служб в одну или добавление средств безопасности в незащищенную службу. Виртуализация может быть приложена или вложена в многоуровневые системы». А вот взятое из того же источника определение виртуального устройства: «Виртуальное устройство может быть представлено операционной системе средствами управляющего программного обеспечения, например, менеджера томов. С точки зрения приложения виртуальное устройство, идентично физическому. В некоторых случаях возможности виртуальных устройств отличаются от физических, в частности, обычно с виртуального устройства невозможно осуществить первичную загрузку». Если попытаться перевести это на обычный язык, можно сказать, что предназначение виртуализации состоит в том, чтобы: изолировать пользователя (сервер или приложение) от физических ресурсов хранения: программа может не знать реальных адресов или параметров диска; изолировать пользователя (сервер или приложение) от изменений в инфраструктуре корпоративного хранилища данных; способствовать организации многопользовательского режима, повышающего эффективность использования ресурсов; помочь администратору управлять большими объемами хранения.

Внедрение виртуализации преследует несколько целей:

– формирование единообразного взгляда на хранение данных вне зависимости от физической природы и топологии систем хранения;

– создание единой точки управления, сосуществующей с аналогичными точками управления серверами, операционными и файловыми системами;

– возможность выбора накопителей, в наибольшей степени соответствующих заданным требованиям QoS и, следовательно, развития и поддержки гетерогенных сетей хранения;

– обеспечение высокой готовности, масштабируемости, безопасности и других эксплуатационных показателей.

Понятие виртуализации условно можно разделить на две фундаментально различающиеся категории:

Виртуализация платформ. Продуктом этого вида виртуализации являются виртуальные машины - некие программные абстракции, запускаемые на платформе реальных аппаратно-программных систем.

Данный вид виртуализации преследует своей целью комбинирование или упрощение представления аппаратных ресурсов для пользователя и получение неких пользовательских абстракций оборудования, пространств имен, сетей и т.п.

Под виртуализацией платформ понимают создание программных систем на основе существующих аппаратно-программных комплексов, зависящих или независящих от них. Система, предоставляющая аппаратные ресурсы и программное обеспечение, называется хостовой (host), а симулируемые ей системы – гостевыми (guest). Чтобы гостевые системы могли стабильно функционировать на платформе хостовой системы, необходимо, чтобы программное и аппаратное обеспечение хоста было достаточно надежным и предоставляло необходимый набор интерфейсов для доступа к его ресурсам. Есть несколько видов виртуализации платформ, в каждом из которых осуществляется свой подход к понятию «виртуализация». Виды виртуализации платформ зависят от того, насколько полно осуществляется симуляция аппаратного обеспечения. До сих пор нет единого соглашения о терминах в сфере виртуализации, поэтому некоторые из приведенных далее видов виртуализации могут отличаться от тех, что предоставят другие источники.

*Виды виртуализации платформ:*

а) полная эмуляция (симуляция). При таком виде виртуализации виртуальная машина полностью виртуализует все аппаратное обеспечение при сохранении гостевой операционной системы в неизменном виде. Такой подход позволяет эмулировать различные аппаратные архитектуры. Например, можно запускать виртуальные машины с гостевыми системами для x86-процессоров на платформах с другой архитектурой (например, на RISC-серверах компании Sun). Долгое время такой вид виртуализации использовался, чтобы разрабатывать программное обеспечение для новых процессоров еще до того, как они станут физически доступными. Такие эмуляторы также применяют для низкоуровневой отладки операционных систем. Основной минус данного подхода заключается в том, что эмулируемое аппаратное обеспечение весьма и весьма существенно замедляет быстродействие гостевой системы, что делает работу с ней очень неудобной, поэтому, кроме как для разработки системного программного обеспечения, а также образовательных целей, такой подход мало где используется. Примеры продуктов для создания эмуляторов: Bochs, PearPC, QEMU (без ускорения), Hercules Emulator;

б) частичная эмуляция (нативная виртуализация). В этом случае виртуальная машина виртуализирует лишь необходимое количество аппаратного обеспечения, чтобы она могла быть запущена изолированно.

Такой подход позволяет запускать гостевые операционные системы, разработанные только для той же архитектуры, что и у хоста. Таким образом, несколько экземпляров гостевых систем могут быть запущены одновременно. Этот вид виртуализации позволяет существенно увеличить быстродействие гостевых систем по сравнению с полной эмуляцией и широко используется в настоящее время. Также, в целях повышения быстродействия, в платформах виртуализации, использующих данный подход, применяется специальная «прослойка» между гостевой операционной системой и оборудованием (гипервизор), позволяющая гостевой системе напрямую обращаться к ресурсам аппаратного обеспечения. Гипервизор, называемый также «Монитор виртуальных машин» (Virtual Machine Monitor) - одно из ключевых понятий в мире виртуализации. Применение гипервизора, являющегося связующим звеном между гостевыми системами и аппаратурой, существенно увеличивает быстродействие платформы, приближая его к быстродействию физической платформы.

К минусам данного вида виртуализации можно отнести зависимость виртуальных машин от архитектуры аппаратной платформы.

Примеры продуктов для нативной виртуализации: VMware Workstation, VMware Server, VMware ESX Server, Virtual Iron, Virtual PC, VirtualBox, Parallels Desktop и другие.

Частичная виртуализация, а также «виртуализация адресного пространства» («address space virtualization»).

При таком подходе, виртуальная машина симулирует несколько экземпляров аппаратного окружения (но не всего), в частности, пространства адресов. Такой вид виртуализации позволяет совместно использовать ресурсы и изолировать процессы, но не позволяет разделять экземпляры гостевых операционных систем. Строго говоря, при таком виде виртуализации пользователем не создаются виртуальные машины, а происходит изоляция каких-либо процессов на уровне операционной системы. В данный момент многие из известных операционных систем используют такой подход. Примером может послужить использование UML (User-mode Linux), в котором «гостевое» ядро запускается в пользовательском пространстве базового ядра (в его контексте).

Паравиртуализация. При применении паравиртуализации нет необходимости симулировать аппаратное обеспечение, однако, вместо этого (или в дополнение к этому), используется специальный программный интерфейс (API) для взаимодействия с гостевой операционной системой. Такой подход требует модификации кода гостевой системы, что, с точки зрения сообщества, Open Source не так и критично. Системы для паравиртуализации также имеют свой гипервизор, а API-вызовы к гостевой системе, называются «hypercalls» (гипервызовы). Многие сомневаются в перспективах этого подхода виртуализации, поскольку в данный момент все решения производителей аппаратного обеспечения в отношении виртуализации направлены на системы с нативной виртуализацией, а

поддержку паравиртуализации приходится искать у производителей операционных систем, которые слабо верят в возможности предлагаемого им средства. В настоящее время провайдерами паравиртуализации являются компании XenSource и Virtual Iron, утверждающие, что быстродействие паравиртуализации выше.

Виртуализация уровня операционной системы. Сутью данного вида виртуализации является виртуализация физического сервера на уровне операционной системы в целях создания нескольких защищенных виртуализированных серверов на одном физическом. Гостевая система, в данном случае, разделяет использование одного ядра хостовой операционной системы с другими гостевыми системами. Виртуальная машина представляет собой окружение для приложений, запускаемых изолированно. Данный тип виртуализации применяется при организации систем хостинга, когда в рамках одного экземпляра ядра требуется поддерживать несколько виртуальных серверов клиентов. Виртуализация уровня операционной системы предоставлены на Рисунке 1.1.

Примеры виртуализации уровня ОС: Linux-VServer, Virtuozzo, OpenVZ, Solaris Containers и FreeBSD Jails.

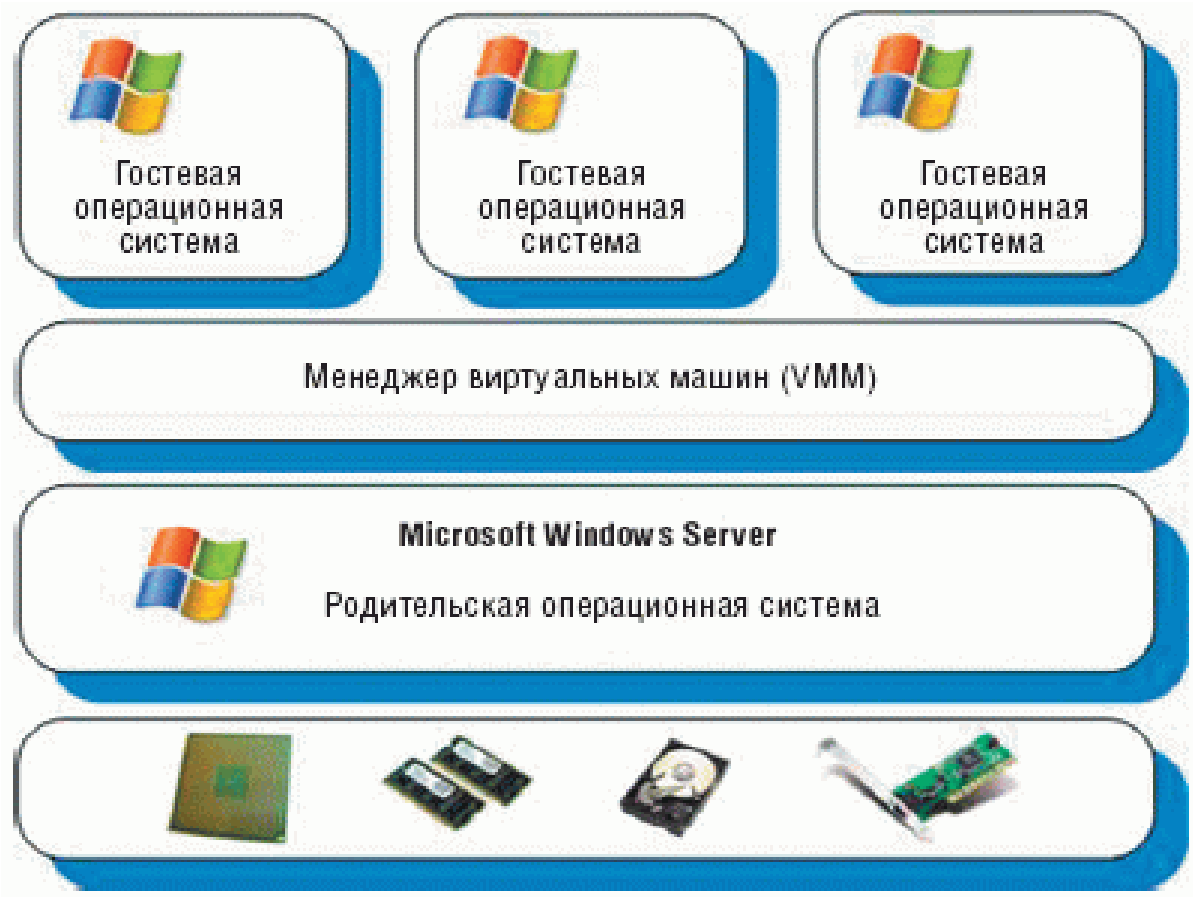


Рисунок 1.1 – Виртуализация уровня операционной системы

Этот вид виртуализации не похож на все остальные: если в предыдущих случаях создаются виртуальные среды или виртуальные

машины, использующиеся для изоляции приложений, то в данном случае само приложение помещается в контейнер с необходимыми элементами для своей работы: файлами реестра, конфигурационными файлами, пользовательскими и системными объектами.

В результате получается приложение, не требующее установки на аналогичной платформе. При переносе такого приложения на другую машину и его запуске, виртуальное окружение, созданное для программы, разрешает конфликты между ней и операционной системой, а также другими приложениями. Такой способ виртуализации похож на поведение интерпретаторов различных языков программирования (недаром интерпретатор, Виртуальная Машина Java (JVM), тоже попадает в эту категорию). Примером такого подхода служат: Thinstall, Altiris, Trigence, Softricity, Microsoft Application Virtualization.

Лидирующими поставщиками виртуализации приложений на рынке являются фирмы Microsoft, Vmware, Citrix. В направлении виртуализации приложений они могут нам предложить следующие продукты:

– Vmware ThinApp;
– Microsoft App-V;
– Citrix XenApp (application streaming).

Если есть продукты, значит, они призваны решать проблемы и оказывать помощь рядовому системному администратору. Итак, какие же проблемы решает виртуализация приложений? Пойдем по порядку:

– снижение расходов на хранение данных. На рабочем месте хранится только кэшированная копия приложений, которые были запущено прежде. К примеру, если из пакета Microsoft Office используется только Microsoft Outlook, то только его кэшированная копия и будет храниться локально;

– уменьшение числа образов персональных компьютеров, которыми необходимо управлять. Так как все приложения хранятся на сервере в запакованных контейнерах, на машине пользователя нет локальных инсталляций программного обеспечения, за счет этого у нас существенно уменьшается количество поддерживаемых образов;

– оптимизация установки обновлений. Сколько времени необходимо на переход с Microsoft Office 2007 на 2010 версию? Без использования технологий виртуализации приложений - 2-часа на каждый персональный компьютер минимум! С технологией виртуализации приложений – 15 минут для всех пользователей;

– использование нескольких версий приложений. Так как приложения запускаются в отдельных контейнерах, никто не мешает запустить Microsoft Office 2007 sp1 и Office 2007 sp2, одновременно на одном ПК. Данный подход будет весьма полезен при разработке программного обеспечения.

Как же работает виртуализация приложений? Для того чтобы начать работать с виртуализацией приложений необходимо:

а) используя Sequencer «завернуть» приложение в контейнер. Программа Sequencer отслеживает и регистрирует процессы установки и запуска программного обеспечения в операционной системе, после этого программа Sequencer формирует пакет виртуализированного приложения и создает виртуальное приложение, которое можно распространять на конечные компьютеры и запускать в собственной виртуальной среде;

б) распространить данное приложение. Способов распространения очень много, начиная от копирования на USB накопитель, заканчивая установкой специализированного сервера для распространения и управления программным обеспечением, как сделано в продуктах Microsoft App-V и Citrix XenApp.

Виртуализация ресурсов. При описании виртуализации платформ мы рассматривали понятие виртуализации в узком смысле, преимущественно применяя его к процессу создания виртуальных машин. Однако если рассматривать виртуализацию в широком смысле, можно прийти к понятию виртуализации ресурсов, обобщающим в себе подходы к созданию виртуальных систем. Виртуализация ресурсов позволяет концентрировать, абстрагировать и упрощать управление группами ресурсов, таких как сети, хранилища данных и пространства имен.

Объединение, агрегация и концентрация компонентов. Под таким видом виртуализации ресурсов понимается организация нескольких физических или логических объектов в пулы ресурсов (группы), представляющих удобные интерфейсы пользователю. Примеры такого вида виртуализации:

– многопроцессорные системы, представляющиеся нам как один мощный компьютер;

– RAID-массивы и средства управления томами, комбинирующие несколько физических дисков в один логический;

– виртуализация систем хранения, используемая при построении сетей хранения данных SAN (Storage Area Network);

– виртуальные частные сети (VPN) и трансляция сетевых адресов (NAT), позволяющие создавать виртуальные пространства сетевых адресов и имен;

– кластеризация компьютеров и распределенные вычисления (grid computing);

– этот вид виртуализации включает в себя техники, применяемые при объединении множества отдельных компьютеров в глобальные системы (метакомпьютеры), совместно решающие общую задачу.

Разделение ресурсов (partitioning). При разделении ресурсов в процессе виртуализации происходит разделение какого-либо одного большого ресурса на несколько однотипных объектов, удобных для использования. В сетях хранения данных это называется зонированием ресурсов («zoning»).

Инкапсуляция. Многим это слово известно как сокрытие объектом внутри себя своей реализации. Применительно к виртуализации, можно сказать, что это процесс создания системы, предоставляющей пользователю удобный интерфейс для работы с ней и скрывающей подробности сложности своей реализации. Например, использование центральным процессором кэша для ускорения вычислений не отражается на его внешних интерфейсах.

Виртуализация ресурсов, в отличие от виртуализации платформ, имеет более широкий и расплывчатый смысл и представляет собой массу различных подходов, направленных на повышение удобства обращения пользователей с системами в целом. Поэтому, далее мы будем опираться в основном на понятие виртуализации платформ, поскольку технологии, связанные именно с этим понятием, являются в данный момент наиболее динамично развивающимися и эффективными.

Где применяется виртуализация? Виртуализация операционных систем за последние три-четыре года очень хорошо продвинулась вперед, как в технологическом, так и в маркетинговом смысле. С одной стороны, пользоваться продуктами виртуализации стало намного проще, они стали более надежными и функциональными, а с другой – нашлось немало новых интересных применений виртуальным машинам. Сферу применения виртуализации можно определить, как «место, где есть компьютеры», однако на данный момент можно обозначить следующие варианты использования продуктов виртуализации:

Консолидация серверов. В данный момент приложения, работающие на серверах в IT-инфраструктуре компаний, создают небольшую нагрузку на аппаратные ресурсы серверов (в среднем 5-15 процентов). Виртуализация позволяет мигрировать с этих физических серверов на виртуальные и разместить их всех на одном физическом сервере, увеличив его загрузку до 60-80 процентов и, повысив тем самым коэффициент использования аппаратуры, что позволяет существенно сэкономить на аппаратуре, обслуживании и электроэнергии. Консолидация серверов предоставлена на Рисунке 1.2.

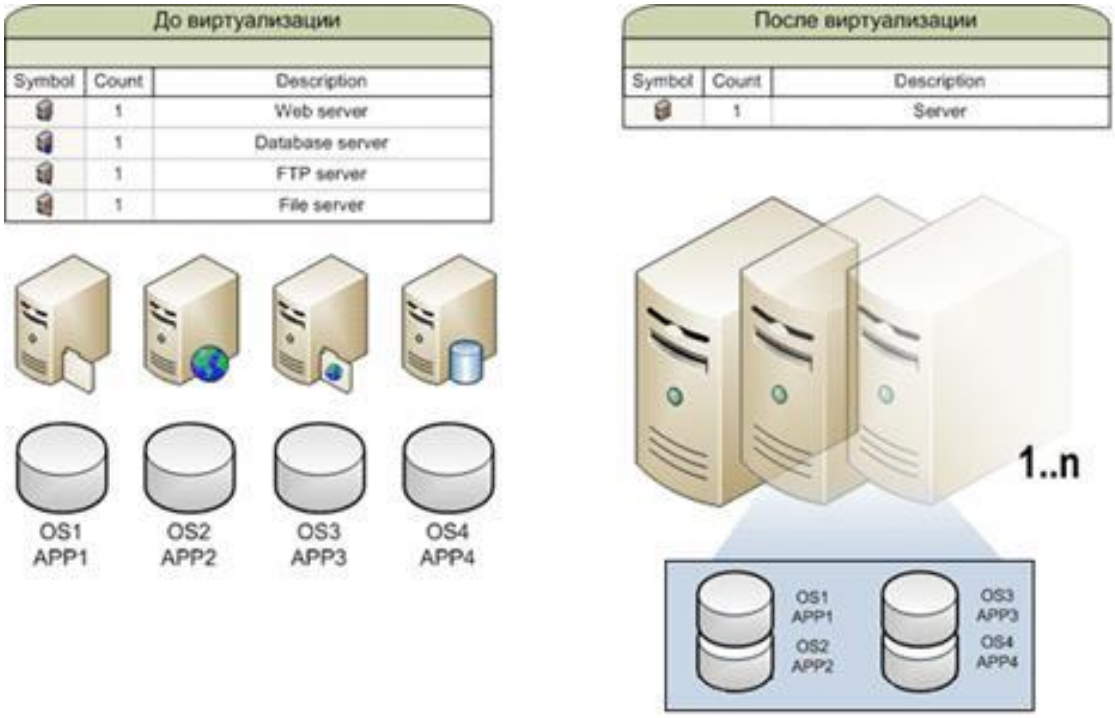


Рисунок 1.2 – Консолидация серверов

Множество продуктов виртуализации позволяют запускать несколько различных операционных систем одновременно, позволяя тем самым разработчикам и тестерам программного обеспечения тестировать их приложения на различных платформах и конфигурациях. Также удобные средства по созданию «снимков» текущего состояния системы одним кликом мыши и такого же простого восстановления из этого состояния, позволяют создавать тестовые окружения для различных конфигураций, что существенно повышает скорость и качество разработки. Предоставление доступа к нескольким виртуальным машинам предоставлена на Рисунке 1.3

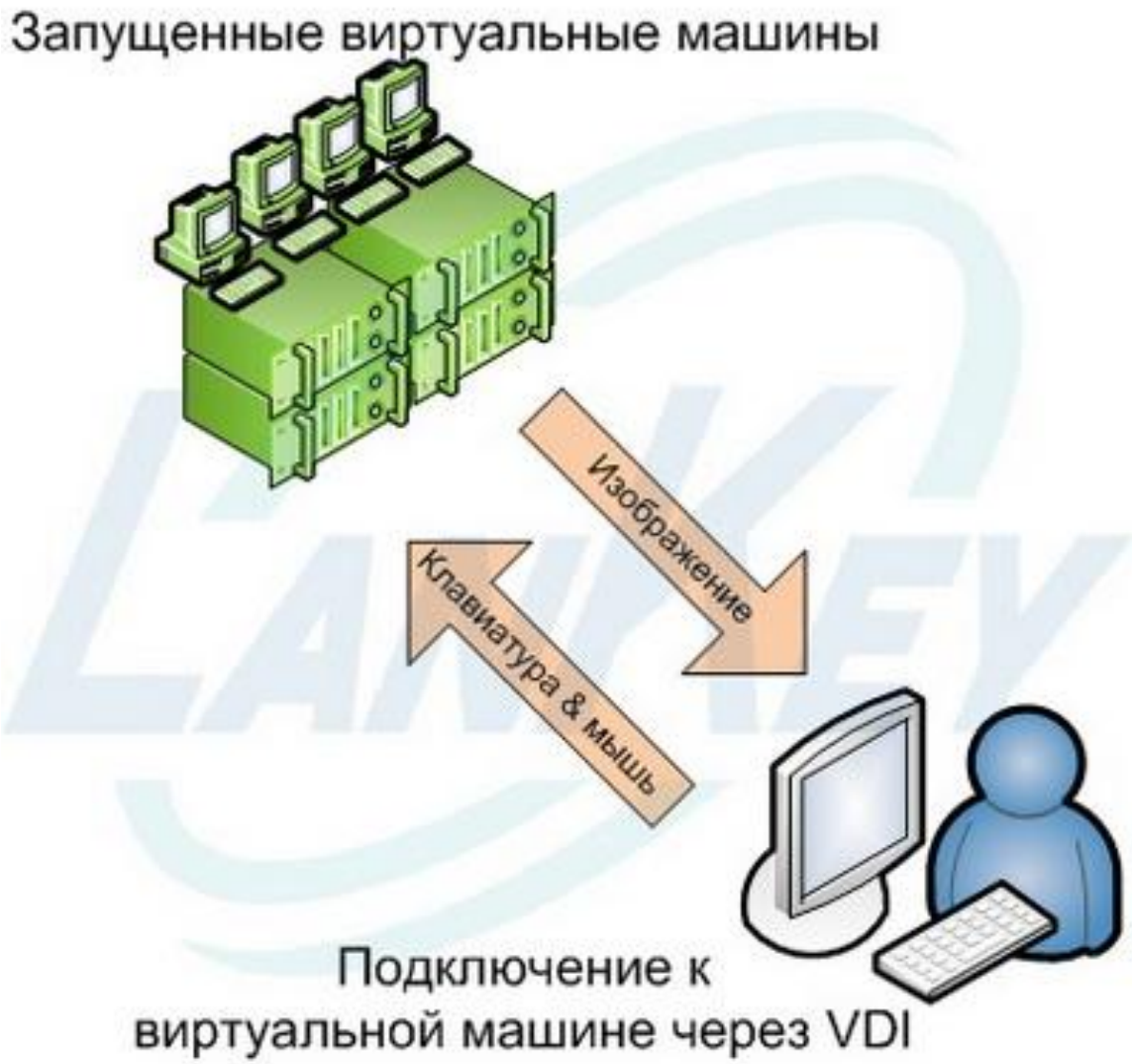


Рисунок 1.3 – Доступ к виртуальным машинам

Использование в бизнесе. Этот вариант использования виртуальных машин является наиболее обширным и творческим. К нему относится все, что может понадобиться при повседневном обращении с IT-ресурсами в бизнесе. Например, на основе виртуальных машин можно легко создавать резервные копии рабочих станций и серверов (просто скопировав папку), строить системы, обеспечивающие минимальное время восстановления после сбоев, и т.п. К данной группе вариантов использования относятся все те бизнес-решения, которые используют основные преимущества виртуальных машин.

Использование виртуальных рабочих станций. С приходом эры виртуальных машин будет бессмысленно делать себе рабочую станцию с ее привязкой к аппаратуре. Теперь создав однажды виртуальную машину со своей рабочей или домашней средой, можно будет использовать её на любом другом компьютере. Также можно использовать готовые шаблоны виртуальных машин (Virtual Appliances), которые решают определенную задачу (например, сервер приложений). Концепция такого использования виртуальных рабочих станций может быть реализована на основе хост-

серверов для запуска на них перемещаемых десктопов пользователей (нечто подобное мэйнфреймам). В дальнейшем эти десктопы пользователь может забрать с собой, не синхронизируя данные с ноутбуком. Этот вариант использования также предоставляет возможность создания защищенных пользовательских рабочих станций, которые могут быть использованы, например, для демонстрации возможностей программы заказчику. Можно ограничить время использования виртуальной машины – и по прошествии этого времени виртуальная машина перестанет запускаться. В этом варианте использования заложены большие возможности. Использование виртуальных рабочих станиций предоставлено на Рисунке 1.4.

Рисунок 1.4 – Использование виртуальных рабочих станций

Технология терминальных серверов. Технология терминальных серверов появилось очень давно, и сейчас по маркетинговым соображениям гордо именуется «виртуализацией представления». В журнале «Системный администратор» было множество статей по терминальным решениям, их возможностям и недостаткам. Вскользь

пройдемся по достоинствам, которые получает системный администратор, внедривший терминальные сервера:

– оптимизация установки обновлений. Так как вы устанавливаете все на один сервер, к которому подключаются тонкие клиенты, обновление и установка нового программного обеспечения у вас занимает минуты;

– увеличение скорости инсталляции новых рабочих мест. Сколько уходит времени на установку рабочей станции для пользователя? Даже при использовании средств автоматизации от часа и более. При использовании терминальных клиентов – 5 – 10 минут, на подключение кабелей и настройку клиента;

– снижение затрат на обслуживание. Как уже много раз говорилось, тонкий клиент это устройство, которое неприхотливо в обслуживании, и совершенно не требует внимания со стороны системного администратора. Есть ли разница в управлении 20 рабочими станциями с полноценными ПК и 20 тонкими клиентами? Да, и она огромная! В случае с терминальными клиентами вам необходимо поддерживать только операционную систему на терминальном сервере;

– лицензирование. Как не крути, а лицензирование тонких клиентов стоит гораздо дешевле, чем лицензирование полноценных рабочих станций. Для примера, цена на подключение к терминальному серверу стоит в районе 90 долларов, цена же на Windows 7 Professional в среднем стоит 190 долларов, что на 100 долларов дороже;

Технология терминальных серверов берет свое начало с мэйнфреймов, и в ней заложены основные принципы тех времен:

– запуск приложения происходит на сервере, но отображение его идет на клиентском рабочем месте;

– по сети передаются образы экрана, нажатия клавиш и движения мыши.

В эпоху Windows 2003 Server к недостаткам терминальных серверов относили невозможность отображения полноценной 3D графики и работы в CAD приложениях. С выходом windows 2008 r2 sp1, с технологией RemoteFX данная проблема решена. Так же для решения проблем работы с графикой можно расширить стандартные терминальные службы Windows Server программным обеспечением от компании Citrix XenApp. Помимо работы с графикой у данного программного обеспечения очень много других полезных качеств и свойств, таких как масштабируемость, управляемость и т.д, которые тянут на еще пару статей об этом продукте. Все это делает Citrix XenApp безусловным лидером на рынке терминальных технологий уже не один год.

Технология Virtual Desktop Infrastructure (VDI). Как и виртуализация приложений, технология VDI является сравнительно новой и еще не завоевавшей популярность среди системных администраторов. Все дело в том, что она сложна и не понятна с первого взгляда. На самом деле все очень просто, пользователю вместо доступа к терминальному серверу

дается доступ к своей собственной операционной системе, да только она находится в центре обработки данных в виде виртуальной машины. Пользователь, получив такую виртуальную машину, может работать точно так же, как и на локальной рабочей станции. Какие же плюсы от всего этого? В отличие от терминальных технологий, VDI обладает такими следующими преимуществами:

– полная изоляция приложений. В отличие от терминальных серверов, приложения исполняются в операционной системе пользователя, поэтому риски несовместимости ПО с терминальным сервером исключены;

– полный контроль ресурсов. Используя технологии виртуализации, мы можем точно выделить ресурсы на каждую из машин пользователей. У нас исчезают проблемы с тем, что одна машина «съедает» все ресурсы сервера;

– каждый пользователь имеет собственное окружение. Пользователь может, как угодно персонализировать свою личную машину.

Все эти плюсы обязательно помогут упростить жизнь системному администратору. Ведь управлять виртуальными машинами в центре обработки данных в разы проще, чем обычными рабочими станциями.

Для понимания отличий терминальных технологий и VDI достаточно взглянуть на Рисунок 1.5-1.6.

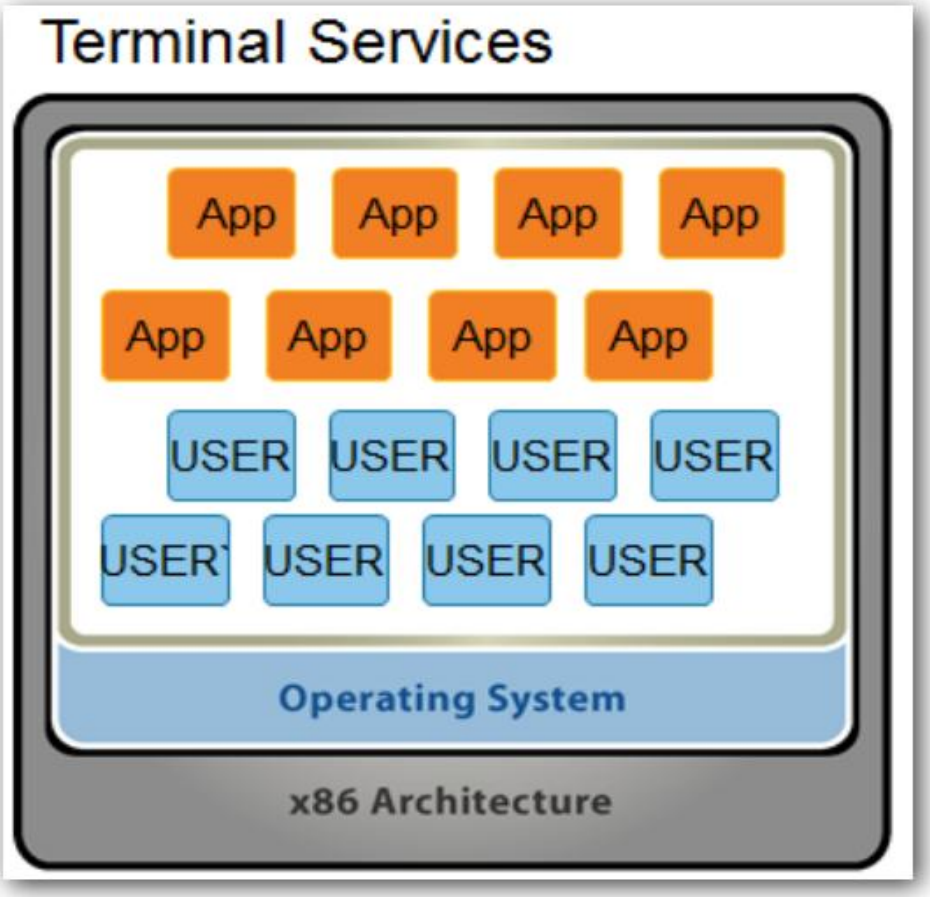


Рисунок 1.5 – Отличие терминальных технологий и Virtual Desktop Infrastructure

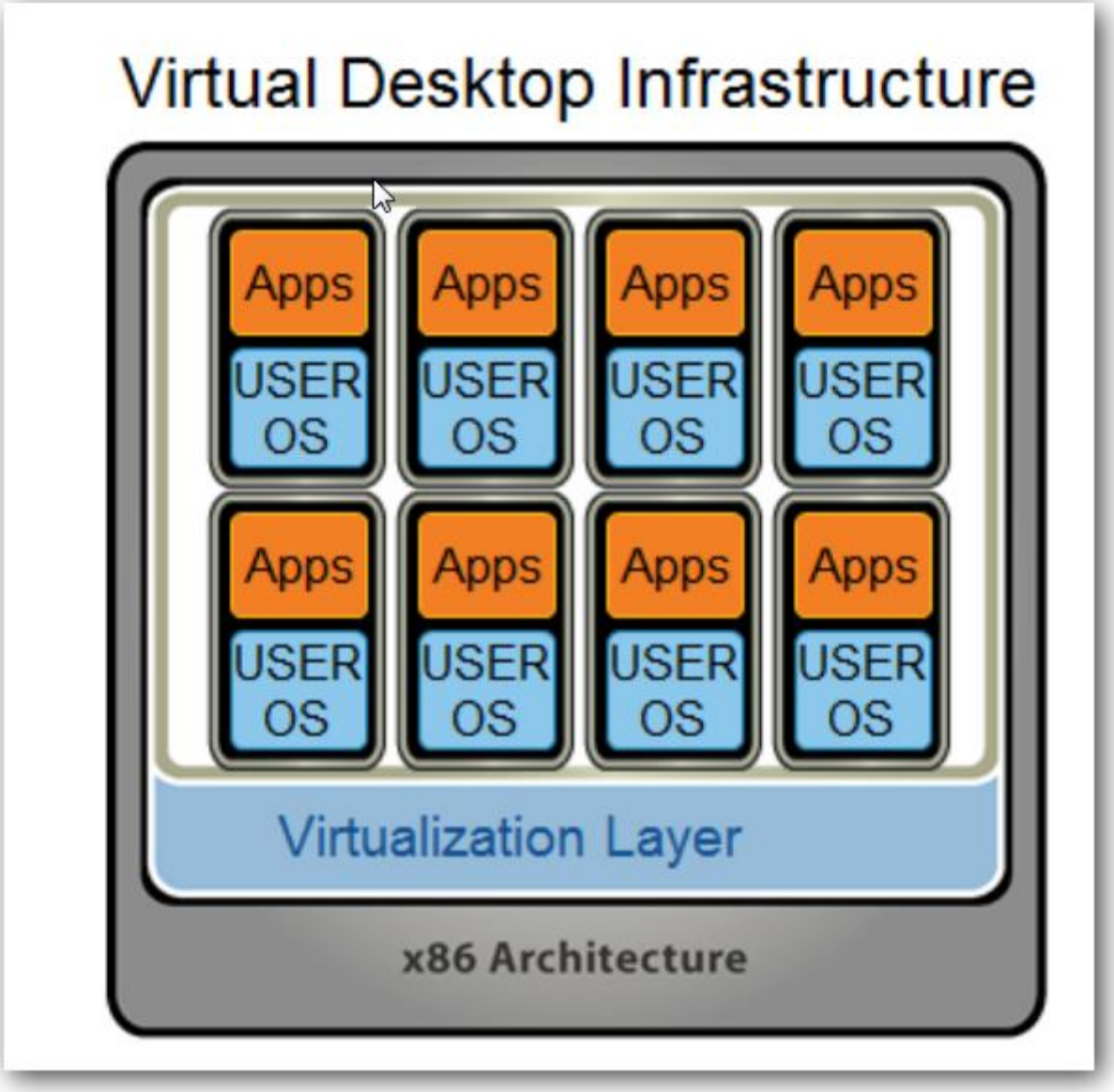


Рисунок 1.6 – Отличие терминальных технологий и Virtual Desktop Infrastructure

Какими же продуктами и вендорами на рынке представлена технология Virtual Desktop Infrastructure? Все те же «три кита» - Microsoft, Citrix, Vmware:

- Citrix XenDesktop;
- Vmware View;
- Microsoft VDI.

Лидером в этой отрасли является компания Citrix, хотя Vmware уже наступает ей на пятки. Microsoft пока находится на позиции догоняющей, хотя с приобретением технологии RemoteFX всѐ может изменится в ближайшее время.

RemoteFX позволяет пользователю комфортно работать по протоколу Удалѐнного рабочего стола (Remote Desktop Protocol, RDP) с видео

высокой чѐткости и прочей сложной графикой в любых форматах — включая Silverlight, Direct3D 9.0c, трѐхмерные модели и, конечно, Windows Aero. . Кроме того, он обеспечивает полную поддержку двухсторонней синхронной аудиопередачи и низкоуровневое подключение любых устройств USB к удалѐнной рабочей станции. Но не всѐ так безоблачно как кажется, технология имеет свои требования к серверному оборудованию:

– процессоры на сервере должны поддерживать Second Level Address Translation (SLAT);

– видеокарта на сервере должна поддерживать DirectX 9.0c и DirectX 10.0 и иметь достаточный объем видеопамяти, которого хватит на выполнение задач во всех виртуальных машинах в зависимости от разрешения экрана;

– если в сервере установлено несколько видеокарт или подключено несколько устройств формата appliance, то такие устройства должны быть на одинаковых графических чипах;

– живая миграция виртуальных машин, в которых используется RemoteFX, требует наличия на серверах одинаковых видеокарт.

Все перечисленные варианты использования виртуальных машин фактически являются лишь сферами их применения в данный момент, со временем, несомненно, появятся новые способы заставить виртуальные машины работать в различных отраслях IT. Но давайте посмотрим, как сейчас обстоят дела с виртуализацией.

## 1.2 Виртуализация сегодня

На сегодняшний день проекты по виртуализации IT-инфраструктуры активно внедряются многими ведущими компаниями, занимающимися системной интеграцией и являющимися авторизованными партнерами провайдеров систем виртуализации. В процессе виртуализации IT-инфраструктуры создается виртуальная инфраструктура – комплекс систем на основе виртуальных машин, обеспечивающих функционирование всей IT-инфраструктуры, обладающий многими новыми возможностями при сохранении существующей схемы деятельности IT-ресурсов. Компании-поставщики различных платформ виртуализации готовы предоставить информацию об успешных проектах по внедрению виртуальной инфраструктуры в крупных банках, промышленных компаниях, больницах, образовательных учреждениях. Множество достоинств виртуализации операционных систем позволяют компаниям экономить на обслуживании, персонале, аппаратном обеспечении, обеспечении бесперебойной работы, репликации данных и восстановлении после сбоев. Также рынок виртуализации начинает наполняться мощными средствами управления, миграции и поддержки виртуальных инфраструктур, позволяющими использовать преимущества виртуализации наиболее полно. Виртуальные

машины открывают широчайшие возможности по переходу ИТ-инфраструктуры компании на новый, более гибкий и технологичный уровень, позволяющий строить ее из независимых, изолированных блоков, которые консолидировано работают без привязки к физическому оборудованию. Назовем основные предпосылки к виртуализации компьютерных систем компании:

– малая средняя загрузка внутренних и внешних (production) серверов (не более 40%);

– отсутствие специфического и обязательного к использованию аппаратного обеспечения;

– необходимость в эффективном управлении большим парком серверов;

– необходимость поддержки бесперебойного режима работы серверов с малым временем простоя (downtime);

– большие затраты на аппаратное обеспечение и электроэнергию в датацентрах;

– необходимость иметь достаточное количество систем для тестирования ПО.

### 1.3 Преимущества и недостатки использования виртуальных машин

Преимущества. Экономия на аппаратном обеспечении при консолидации серверов.

Существенная экономия на приобретении аппаратного обеспечения происходит при размещении нескольких виртуальных серверов на одном физическом. В зависимости, от компании-поставщика платформы виртуализации, доступны возможности по балансировке рабочей нагрузки, контролю выделяемых ресурсов, миграции между физическими хостами и бэкапу. Все это влечет за собой реальную экономию денежных средств на обслуживании, управлении и администрировании инфраструктуры серверов.

Не нужно менять кулеры на процессорах, ремонтировать вышедший из строя блок питания и т.д. и т.п. С другой стороны повышается требовательность к аппаратным ресурсам (минус). В случае размещения на одном физическом и множестве виртуальных серверов различных ОС и приложений мы выигрываем следующее:

а) оборудование для системы занимает меньше места в стойке. Вместо стойки в 3-4 маломощных сервера можно установить один сервер, который будет выполнять те же самые функции. Особенно это критично, если оборудование размещается на арендуемой площадке провайдера или не хватает места в стойке. Для виртуальных машин, размещенных на одном сервере, требуется меньше портов на KVM-switch (Keyboard-Video-Mouse);

б) добавляются дополнительные возможности администрирования. Например, появляется возможность при зависании приложения не только перезапустить нужные процесс, а перезагрузить виртуальный сервер целиком, не влияя на работу приложений, размещенных на других виртуальных серверах.

Виртуальные машины могут создавать представления устройств, которых у вас нет.

Например, многие системы виртуализации позволяют создавать виртуальные SCSI диски, виртуальные многоядерные процессоры и т.п. Это может пригодиться для создания различного рода симуляций.

На одном хосте может быть запущено одновременно несколько виртуальных машин, объединенных в виртуальную сеть.

Такая особенность предоставляет безграничные возможности по созданию моделей виртуальной сети между несколькими системами на одном физическом компьютере. Особенно это необходимо, когда требуется смоделировать некую распределенную систему, состоящую из нескольких машин. Также можно создать несколько изолированных пользовательских окружений (для работы, развлечений, работы в Интернет), запустить их и переключаться между ними по мере необходимости выполнения тех или иных задач.

Виртуальные машины повышают мобильность. Папка с виртуальной машиной может быть перемещена на другой компьютер, и там виртуальная машина может быть сразу запущена. Не требуется создавать никаких образов для миграции, и, к тому же, виртуальная машина отвязана от конкретной аппаратуры.

Виртуальные машины более управляемы. При использовании виртуальных машин существенно повышается управляемость в отношении создания резервных копий, создания снимков состояний виртуальных машин («снэпшотов») и восстановлений после сбоев в связи с тем, что серверы служб получают независимость от железа, на котором работают.

На этом, конечно, достоинства виртуальных машин не исчерпываются, это лишь пища для размышления и исследования их возможностей. Безусловно, как и у всякого нового и перспективного решения, у виртуальных машин есть и свои недостатки:

Невозможность эмуляции всех устройств. В данный момент все основные устройства аппаратных платформ поддерживаются вендорами систем виртуализации, однако если вы используете, например, какие-либо контроллеры или устройства, не поддерживаемые ими, придется отказаться от виртуализации такого окружения.

Виртуализация требует дополнительных аппаратных ресурсов. В настоящее время использование различных техник виртуализации позволило приблизить показатели быстродействия виртуальных машин к реальным, однако, чтобы физический хост смог запускать хотя бы пару виртуальных машин, требуется достаточное для них количество аппаратных ресурсов.

Некоторые платформы виртуализации требовательны к конкретному аппаратному обеспечению.

В частности, замечательная платформа компании VMware, ESX Server, была бы и вовсе замечательной, если бы не предъявляла жестких требований к аппаратному обеспечению.

Хорошие платформы виртуализации стоят хороших денег. Порой, стоимость развертывания одного виртуального сервера равна стоимости еще одного физического, в определенных условиях это может оказаться нецелесообразным. К счастью, есть множество бесплатных решений, но они, в основном, ориентированы на домашнего пользователя и малый бизнес.

Основные проблемы проектов по виртуализации. Несмотря на все преимущества виртуализации, при проведении проекта по ее внедрению компании различного масштаба сталкиваются с несколькими проблемами, которые появляются на различных этапах проекта внедрения. Условно можно выделить три фазы, в которых появляются трудности с виртуализацией:

- анализ и планирование;
- внедрение и период поствнедрения;
- дальнейшее обслуживание виртуальной инфраструктуры.

Анализ и планирование. В эту группу включаются проблемы, с которыми компания или независимые консультанты сталкиваются при исследовании существующей информационной инфраструктуры и планировании миграции на виртуальную инфраструктуру. В частности, среди первоочередных задач можно отметить следующие:

Совместимость и поддержка. Эта проблема содержит исследование аппаратных и программных компонентов ИТ-инфраструктуры и оценку возможности их миграции в виртуальные среды. Не все разработчики программного обеспечения готовы гарантировать бесперебойную работу своего ПО в виртуальных машинах. Для решения этой проблемы необходимо провести тщательную инвентаризацию программного и аппаратного обеспечения и выяснить у разработчиков, насколько гарантируется их работа в пределах виртуальной инфраструктуры. Сейчас многие производители платформ виртуализации публикуют различные отчеты о совместимости и тестировании производительности популярных программных продуктов на виртуальных машинах в производственной среде. Например, в июле 2007 года компания VMware опубликовала отчет о производительности продукта Microsoft Exchange на платформе ESX Server, из которого можно сделать вывод, что почтовый сервер Exchange вполне может быть виртуализован с учетом дальнейшего увеличения количества его клиентов и малых затрат на поддержку платформы виртуализации.

Проблемы категории поддержки виртуализации можно разделить на три типа:

– технические ограничения (поддержка в виртуальных машинах специфического оборудования, требования платформы виртуализации к аппаратному обеспечению);

– маркетинговые стратегии (например, решение поддерживать только определенный набор программных продуктов);

– политические стратегии (решения о подходе к виртуализации в контексте поддерживаемого ПО, программная или аппаратная виртуализация, решения использовать платформы определенного вендора).

Лицензирование. Планируя проект по виртуализации, необходимо тщательно изучить лицензии производителей операционных систем, а также независимых разработчиков программного обеспечения (ISV, Independent Software Vendors) в отношении виртуализации. Не исключено, что некоторые ОС или программные продукты не могут быть запущены в виртуальных системах, как, например, Windows Vista Home Basic или Home Premium компании Microsoft.

Ввиду очень простой переносимости виртуальных машин на другую физическую платформу, со стороны производителей операционных систем вводятся специальные ограничения на использование их продуктов в виртуальных машинах (особенно OEM-версий). Такие сценарии использования зачастую описываются в отдельных главах лицензий. Также возникают некоторые проблемы с лицензированием ПО на определенное число процессоров, поскольку в виртуальной машине посредством виртуального SMP (Symmetric Multi Processing) представляется другое число процессоров, нежели в хостовой системе. Кроме того, необходимо учитывать и некоторые приятные моменты в отношении виртуализации операционных систем, как, например, разрешение использовать неограниченное число экземпляров виртуальных систем издания Windows Server 2003 Datacenter Edition без дополнительной платы.

Планирование развертывания. В эту группу задач включается планирование развертывания консолидированных виртуальных серверов, миграция физических серверов, а также получение коэффициента виртуализации (число виртуальных машин на физический сервер). Эта задача является наиболее значимой на этапе анализа и планирования виртуальной инфраструктуры. Некоторые производители платформ виртуализации предоставляют потенциальным клиентам инструменты для оценки коэффициента виртуализации (например, VMware Capacity Planner), либо им приходится пользоваться сторонним ПО (стоящим, как правило, немалых денег), например, PowerRecon компании PlateSpin.

Ниша программного обеспечения для централизованного сбора информации о загрузке серверов еще не заполнена, и на данный момент достаточно сложно подобрать подходящий инструмент.

В отношении миграции физических систем на виртуальные в данный момент существует множество инструментов производителей платформ виртуализации и сторонних разработчиков, которые позволяют производить потоковую миграцию физических серверов. Многие системы

виртуализации уже содержат средства для P2V (Physical to Virtual) миграции (например, Xen Server).

Примеры специализированных продуктов для P2V миграции:

- VMware Converter;
- PlateSpin PowerConvert;
- P2V Direct.

Подготовка персонала. Эта проблема является одной из самых насущных в настоящее время, ввиду отсутствия специалистов по виртуализации, которые смогут внедрить и поддерживать виртуальную инфраструктуру. «Тяжелые» платформы виртуализации (VMware ESX, Xen Enterprise, Virtual Iron) требуют серьезной подготовки специалистов, которые будут поддерживать их в состоянии полной готовности. Обучение таких специалистов обходится довольно дорого и доступно далеко не везде. Отсутствие специалистов необходимого уровня входит в тройку главных причин отказа компаний внедрять виртуализацию.

Оценка возврата инвестиций (ROI, Return of Investments). Как показала практика внедрения систем виртуализации, невозможность грамотно оценить возвращение инвестиций в виртуализацию является главным фактором, из-за которого почти половина компаний не может назвать свои виртуализационные проекты успешными. На данный момент существует проблема со средствами для измерения количественных и качественных показателей виртуальной инфраструктуры. Несмотря на все достоинства виртуализации, сложно говорить о ее эффективности пока нет подобных средств. Тем не менее, нужно четко осознавать, для каких целей нужна виртуализация, и какой именно экономический эффект она принесет.

Внедрение и период пост внедрения. На этапе внедрения платформ виртуализации, как правило, возникают проблемы интеграции виртуальной инфраструктуры с существующими гетерогенными элементами реальной ИТ-инфраструктуры, а также необходимость в развертывании специализированных серверов для управления виртуальными системами, мониторинга загрузки, выделение серверов для резервного копирования, внедрении решений высокой доступности. Кроме того, на данной стадии, при внедрении виртуализации в крупных компаниях, появляются проблемы с использованием виртуальных платформ в сетях SAN. Не все платформы поддерживают большой набор применяемого оборудования и популярный протокол iSCSI. Для внедрения решений виртуализации в сетях хранения данных без специально обученных специалистов не обойтись. Также на этапе внедрения и первичного использования возникают следующие проблемы:

Надежность. Поскольку в результате внедрения виртуализации несколько виртуальных серверов консолидировано работают на одном физическом сервере, необходимо спланировать и реализовать стратегии резервного копирования и восстановления после сбоев (Disaster Recovery). Отказы аппаратных компонентов серверов - не редкость, и платформа с интегрированными средствами для обеспечения надежности виртуальной

инфраструктуры будет лучшим решением. Большим плюсом станет также возможность платформы интегрироваться с решениями других производителей программных продуктов для обеспечения надежности. Резервное копирование виртуальных машин может проводиться на трех уровнях:

– уровень гостевой системы (установка специальных агентов в гостевой системе, например, продукты компании Symantec);

– уровень хостовой системы (копирование образов виртуальных машин или файлов гостевой системы, например, средствами продукта esxRanger компании Vizioncore);

– уровень систем хранения данных (SAN-хранилищ, где блоки виртуальной инфраструктуры соответствуют определенным сегментам).

Обеспечение высокой доступности (HA, High Availability) виртуальных машин может обеспечиваться средствами самих платформ (например, VMware HA), которые содержат несколько элементов (оптимизированная файловая система, возможности автоматического перезапуска виртуальных машин на другом сервере при выходе из строя одного из серверов кластера). В ближайшее время ожидается развитие средств высокой доступности виртуальных машин сторонних производителей. Сейчас в этом направлении активно работают следующие компании:

– Invirtus;
– PlateSpin;
– Symantec;
– Vizioncore.

Развертывание и подготовка к использованию в производственной среде.

Простота развертывания виртуальных машин, зачастую, рождает соблазн их бесконтрольного размещения на различных серверах, что влечет за собой проблемы контроля производительности и распределения загрузки. Довольно сложно адекватно оценить, как именно виртуальные машины воздействуют на сервер виртуализации, и сколько виртуальных машин на нем можно разместить в пределах пула ресурсов. В этом отношении производителями программного обеспечения прилагаются усилия по созданию комплексных систем по управлению развертыванием виртуальных машин и их учету. Наибольших успехов в этой области добились компании:

– Dunes (с продуктом Virtual Desktop Orchestrator);
– VMware (с продуктом VMware LabManager);
– Surgient (с серией продуктов VQMS, VTMS, VDMS);
– VMLogix (с продуктом для тестовых лабораторий VMLogix Lab Manager).

Оценка эффективности. После того как виртуальные машины будут развернуты, необходимо оценить их эффективность в различных аспектах.

В виртуальных инфраструктурах процесс выявления узких мест в производительности виртуальных машин происходит значительно сложнее, чем в реальных. Наличие «затычек» производительности зависит не только от стандартных для этой проблемы факторов, но еще и от некоторых специфических моментов (эффективность эмуляции аппаратных компонентов, проблемы производительности хостовой платформы, настройка параметров платформы виртуализации и прочее). Нахождение и изоляция такого рода узких мест (bottlenecks) – задача особого класса программного обеспечения. Поставщики ПО для оценки производительности компьютерных систем в недалеком будущем адаптируют продукты для платформ виртуализации. Кроме того, некоторые вендоры систем виртуализации уже сейчас предлагают подобные инструменты. Компания VMware недавно выпустила продукт VMmark для комплексной оценки производительности виртуальных машин на платформе ESX Server.

Как видно из отчета, система в каждом из аспектов набирает определенное количество очков, которое в дальнейшем может использоваться производителями аппаратных компонентов для рекомендаций по применению своего «железа» в системах виртуализации. Эти очки могут использоваться также в специальном ПО для планирования виртуальной инфраструктуры. В настоящее время производители аппаратного обеспечения, такие как Intel, активно сотрудничают с поставщиками платформ виртуализации в этом отношении.

Дальнейшее обслуживание виртуальной инфраструктуры. По мере ввода виртуальной инфраструктуры в производственную эксплуатацию перед системными администраторами и аналитиками встают вопросы о ее масштабировании, обслуживании, основанном на централизованном управлении, безопасности и разделения зон ответственности.

Масштабируемость. После развертывания виртуальной инфраструктуры, по мере роста потребностей компании в виртуальных машинах возникает проблема масштабирования парка виртуальных машин. Основной составляющей проблемы является тот факт, что рост мощностей аппаратного обеспечения происходит неравномерно. Например, развитие многопроцессорных систем идет большими темпами, что позволяет увеличивать число виртуальных систем на одной физической. Однако некоторые другие аппаратные компоненты растут не так быстро: типичный пример – пропускная способность компонентов передачи данных по сети. Безусловно, вскоре мы сможем размещать на одном физическом сервере несколько десятков виртуальных машин, запущенных одновременно, но как справляться с их сетевым трафиком?

Кроме того, большие масштабы виртуальной инфраструктуры требуют развертывания в виртуальных машинах однотипного программного обеспечения, управление которым происходит на уровне гостевой системы, что может вызвать определенные трудности. В данный момент только компания VMware работает над тем, чтобы передать

управление развертыванием и обслуживанием ПО на уровень хостовой платформы.

Безопасность. Проблема безопасности виртуальной инфраструктуры может быть условно разделена на две составляющие:

- безопасность виртуальных машин;
- безопасность платформы виртуализации.

В первом случае, так же, как и на физической платформе, необходима установка средств защиты в гостевой операционной системе (антивирусы, сетевые экраны и прочее), а также правильная настройка виртуальных машин и виртуальных сетей. В мире виртуализации могут существовать несколько иные понятия о безопасности, в связи с тем, что еще не окончательно оформились требования к защите виртуальных систем. Например, в некоторых продуктах VMware (Workstation, Server) виртуальные коммутаторы являются по своей природе «хабами», что может открыть возможности по прослушиванию незащищенного трафика другими виртуальными машинами, контролируемыми злоумышленниками, и ввести в заблуждение системных администраторов.

В отношении безопасности платформы виртуализации, при планировании стратегии развертывания виртуальных машин, необходимо проанализировать отчеты об уязвимостях платформ и учитывать этот фактор при выборе конкретной платформы. Частью обслуживания виртуальной инфраструктуры должна стать своевременная установка обновлений системы виртуализации и отслеживание критических уязвимостей. При этом необходимо также учитывать возможность внутреннего несанкционированного доступа в пределах хостовой системы. Сейчас атаки на виртуальные системы достаточно редки, ввиду не такого широкого распространения платформ виртуализации, поэтому истинная защищенность платформ виртуализации неизвестна. Аналитики предсказывают, что к 2015 году каждая вторая операционная система будет виртуализована, и со временем вопрос безопасности будет становиться все более серьезным.

Ответственность: не исключено, что внедрение виртуальной инфраструктуры породит появление новых ролей, полномочия которых необходимо будет делегировать существующим специалистам (системным администраторам, сетевым администраторам, сотрудникам служб поддержки, специалистам по безопасности). Это требует определения четкой модели эксплуатации виртуальных систем в соответствии с функциями каждого из относящихся к этому работников компании. При разделении зон ответственности, необходимо четко представлять варианты использования виртуальных систем, иначе могут возникнуть конфликты между работниками, из-за появления новых обязанностей, исполнение которых никто не захочет брать на себя.

Оценка рынка виртуализации. В данный момент на рынке виртуализации происходит бурный рост, что вызывает серьезную конкурентную борьбу вендоров платформ виртуализации за поддержку их

продуктов разработчиками программного и аппаратного обеспечения. Например, в данный момент компания VMware является абсолютным лидером рынка корпоративного сектора виртуализации, не испытывая особенного сопротивления со стороны компаний Microsoft, XenSource, SWSoft и Virtual Iron. Однако ситуация меняется: в 2012 году вышла интегрированная в ОС Windows Server 2012 платформа виртуализации Microsoft, что может привести к изменению ориентиров рынка, учитывая сеть партнеров софтверного гиганта. Поэтому необходимо внимательно следить за поддержкой продуктов виртуализации различными производителями ПО и делать ставку на конкретную платформу с учетом перспектив развития рынка.

Распределение доль рынка виртуализации предоставлено на Рисунке 1.7.

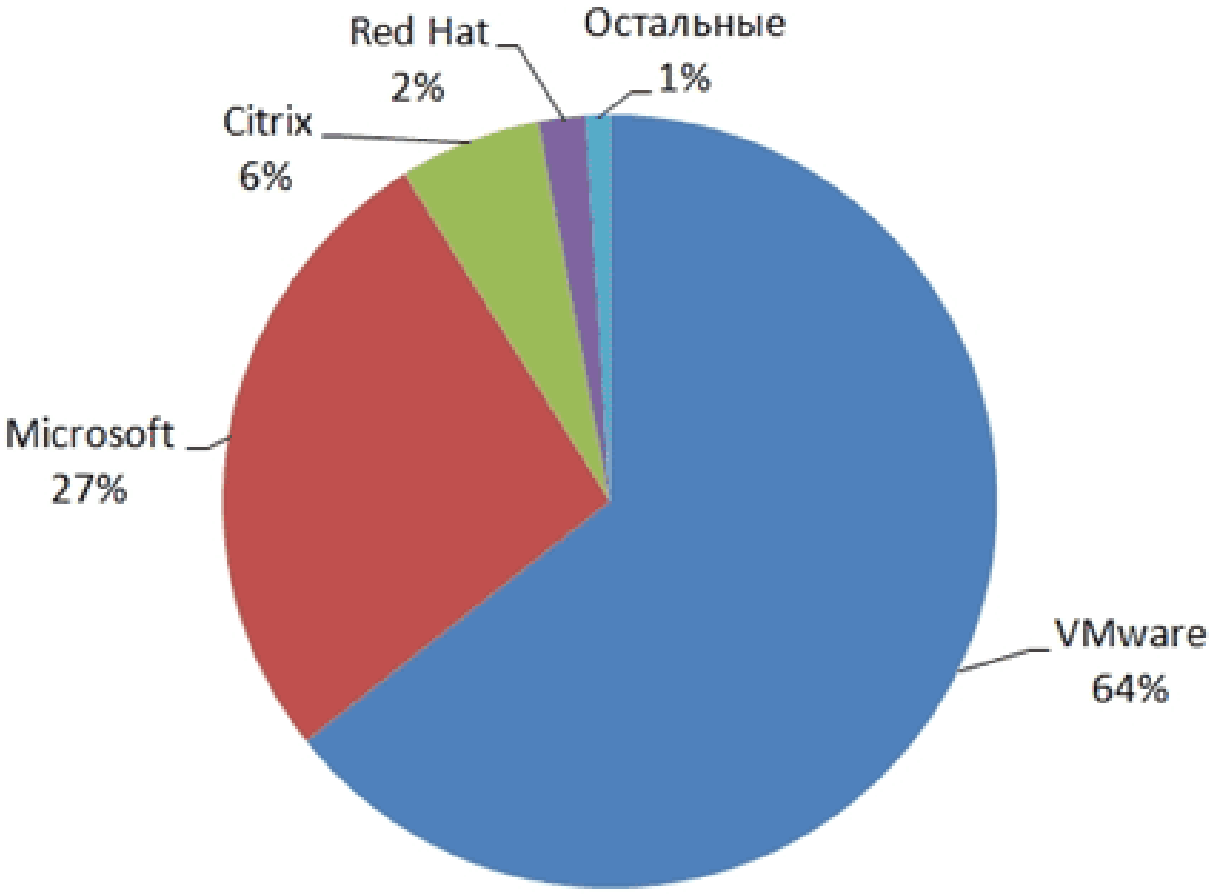


Рисунок 1.7 – График рынка виртуализации

Виртуализация приложений. При виртуализации приложений происходит отделение уровня конфигурации приложения от ОС. Приложения могут работать на клиентских компьютерах, включая настольные ПК, серверы и переносные компьютеры. Эти приложения не

требуется устанавливать, а управлять ими можно из единого центра. Эти особенности имеют огромное значение для выполнения самых разных задач, от управления установкой исправлений и обновлений до развертывания и удаления приложений.

Каждое приложение зависит от своей ОС по ряду причин, включая распределение памяти, драйверы устройства и много другое. Проблемы несовместимости приложений и операционных систем могут быть решены за счет виртуализации сервера или представлений. Однако для устранения проблем несовместимости двух приложений, установленных в одной копии ОС, потребуется выполнить виртуализацию приложений.

Приложения, установленные на одном устройстве, обычно используют общие элементы конфигурации, однако это совместное использование может привести к определенным проблемам. Например, для одного приложения может потребоваться определенная версия библиотеки DLL, тогда как для другого приложения в этой системе будет нужна другая версия той же библиотеки. При установке обоих приложений создается ситуация, в которой одно из них перезаписывает версию библиотеки, необходимую для другого, что приводит к ошибкам приложения или сбою системы. Чтобы этого избежать, организации обычно выполняют всестороннее тестирование нового приложения перед его установкой. Этот подход вполне работоспособен, но дорог и отнимает много времени.

Виртуализация приложений решает эту проблему за счет создания копий всех общих ресурсов, необходимых приложению. Общие элементы конфигурации (разделы реестра, отдельные библиотеки DLL и другие объекты), которые приложение обычно может использовать совместно с другими приложениями системы, «упаковываются» вместе с ним и обрабатываются в кэше компьютера, благодаря чему создается виртуальное приложение. Развернутое виртуальное приложение использует собственную копию этих общих ресурсов.

В обычной вычислительной среде приложения устанавливаются непосредственно в операционной системе. Так как все они записывают данные в общие системные файлы, часто возникают конфликты приложений, приводящие к сбоям и неустойчивой работе.

При виртуализации приложений каждое приложение работает в своей собственной защищенной среде выполнения, которая изолирует его от других приложений и от основной ОС.

Виртуализация приложений значительно упрощает процесс развертывания. Поскольку приложения больше не конфликтуют с версиями библиотек DLL или других общих ресурсов среды, необходимость тестирования новых приложений на наличие конфликтов с существующими приложениями сводится к минимуму. Виртуальные приложения могут работать рядом с обычными установленными приложениями, поэтому нет необходимости выполнять общую виртуализацию. Однако виртуализация помогает избежать многих проблем и сокращает время на обращение пользователей в службу поддержки. Эффективное решение виртуализации

приложений позволяет управлять виртуальными и установленными приложениями с помощью общего интерфейса.

Виртуальные приложения SoftGrid. Виртуализация приложений позволяет запускать приложения локально и экономить ресурсы сервера, устраняет единые точки отказа и снимает ограничения на работу в сети. Решение для виртуализации приложений SoftGrid обеспечивает аналогичную «песочнице» среду под названием SystemGuard, в которой приложения могут выполняться на компьютерах пользователей без локальной установки операционной системы.

Виртуальная среда SystemGuard. SystemGuard — виртуальная прикладная среда, которая содержит все необходимые приложению элементы, такие как файлы, данные реестра, COM-объекты и информация о среде. Совместимые с SoftGrid программы не требуется устанавливать в основной операционной системе, но они взаимодействуют с операционной системой в соответствии с определенными правилами, чтобы избежать излишнего дублирования данных в виртуализованном пространстве.

Как показано на экране 1, приложение взаимодействует с ресурсами, которые представляются ей обычной операционной системой, с полными правами записи и чтения в среде SystemGuard. SystemGuard строго контролируемо взаимодействует с настоящей операционной системой. Информацию о настроечных параметрах можно прочитать, но не изменить. Данные профиля и документов могут быть изменены в операционной системе, которая позволяет сохранять данные и параметры среды между сеансами.

Виртуальная среда состоит из нескольких виртуальных элементов для областей операционной системы, используемых приложениями. Если приложение пытается прочитать что-либо из реестра, а данные реестра в виртуальном реестре отсутствуют, запрос чтения передается операционной системе (экран 2). Запросы записи всегда поступают в виртуальный реестр. Тот же самый процесс применяется для виртуальной файловой системы. Например, виртуальный слой гарантирует, что библиотеки DLL, используемые приложением, всегда считываются сначала из среды SystemGuard. Это позволяет избежать конфликтов с локальными версиями. Если приложению необходима служба, то компонент Virtual Services обеспечивает функционирование службы в SystemGuard незаметно для любого другого приложения, выполняемого в операционной системе. Существуют также виртуальные подсистемы для среды COM, .ini-файлов, ресурсов процесса и шрифтов.

Дополнительные требования к ресурсам. SystemGuard создает меньшую нагрузку на ресурсы, чем может показаться. Помимо дискового пространства для кэширования приложения, которое обычно меньше, чем для локально установленного приложения, виртуализованное приложение потребляет менее чем на 1% больше ресурсов процессора, нежели невиртуализованное приложение. Причина в том, что основная часть дополнительной нагрузки приходится на инициализацию SystemGuard и

открывание приложения, а не на этап его выполнения. Кроме того, потребность в памяти при выполнении виртуализованных приложений, как правило, ниже, чем при выполнении обычных.

Занимаемую приложением память можно разделить на выгружаемый пул и невыгружаемый пул. Большинство приложений работают внутри выгружаемого пула памяти. Содержимое выгружаемого пула можно при необходимости записать на диск, а память, используемая виртуализованным приложением, такая же, как для локально установленных программ. Единственное дополнительное требование к памяти — 20 Мбайт для клиента SoftGrid. Невыгружаемый пул используется для важных данных операционной системы, которые нельзя сохранить на диске.

Настроечные данные (например, данные реестра) обычно загружаются в ядро во время начального запуска компьютера. Когда загружаются программы, реестр растет и занимает больше места, что приводит к увеличению времени загрузки (для областей реестра HKEY_LOCAL_MACHINE и HKEY_CLASSES) или процедур регистрации (HKEY_CURRENT_USER).

Для приложений SoftGrid никакие данные в реестр компьютера не записываются. Данные, необходимые для приложения, загружаются по мере необходимости в процессе выполнения. Представьте, что локально установлено 40 программ, но используются только 10: реестр серьезно разрастается после заполнения невыгружаемого пула. При запуске виртуализованных приложений экономится 75% невыгружаемого пула.

Экономятся и ресурсы сетевого канала связи, так как SoftGrid доставляет клиенту виртуализованные приложения и компоненты по требованию. Так же при виртуализации рабочих мест нет необходимости пересылать «тяжелый» контент приложений по каналам связи. Терминальная сессия тонкого клиента использует менее 256 Кбит/с трафика. Схема подключения тонких клиентов предоставлена на Рисунке 1.8.

Рисунок 1.8 – Схема подключения тонких клиентов

Сначала имеется лишь пиктограмма приложения на рабочем столе клиента. Когда приложение используется, SoftGrid последовательно извлекает компоненты приложения, значительно ускоряя его запуск. Доставляются только нужные компоненты приложения, что экономит место на диске клиента.

Еще одно интересное преимущество состоит в том, что при использовании SoftGrid с серверами терминалов не требуется «терапевтическая перезагрузка». SoftGrid отличается чрезвычайной компактностью при выполнении и аккуратно удаляет «мусор» после закрытия приложения.

Компоненты SoftGrid. Комплекс SoftGrid состоит из пяти основных компонентов, которые позволяют создавать приложения SystemGuard и размещать их на клиентах.

System Center Virtual Application Server. Обеспечивает потоковую передачу SoftGrid-совместимых приложений клиентам и проверяет наличие у клиента прав запуска приложения.

SoftGrid Data Store. Хранит данные о среде SoftGrid. Схема подключения через систему SoftGrid предоставлена на Рисунке 1.9.

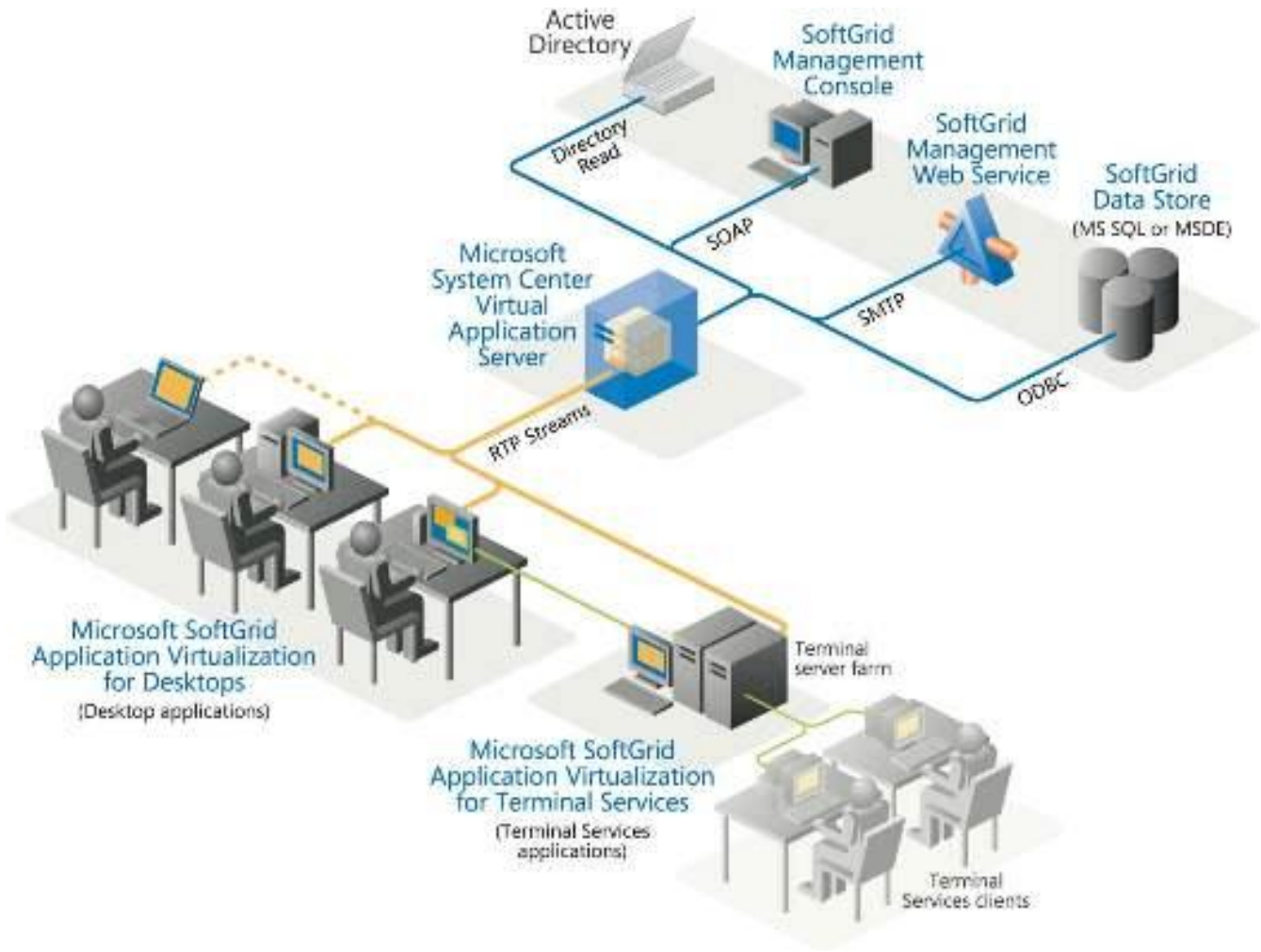


Рисунок 1.9 – Схема подключения тонких клиентов через систему SoftGrid

SoftGrid Management Web Service. Интерфейс между консолью SoftGrid Management Console и хранилищем SoftGrid Data Store. Использует Microsoft .NET Framework 1.1 и более поздние версии. Необходимы Microsoft IIS 5.0 или более поздняя версия и расширения ASP .NET. Схема подключения через систему SoftGrid предоставлена на Рисунке 1.10.

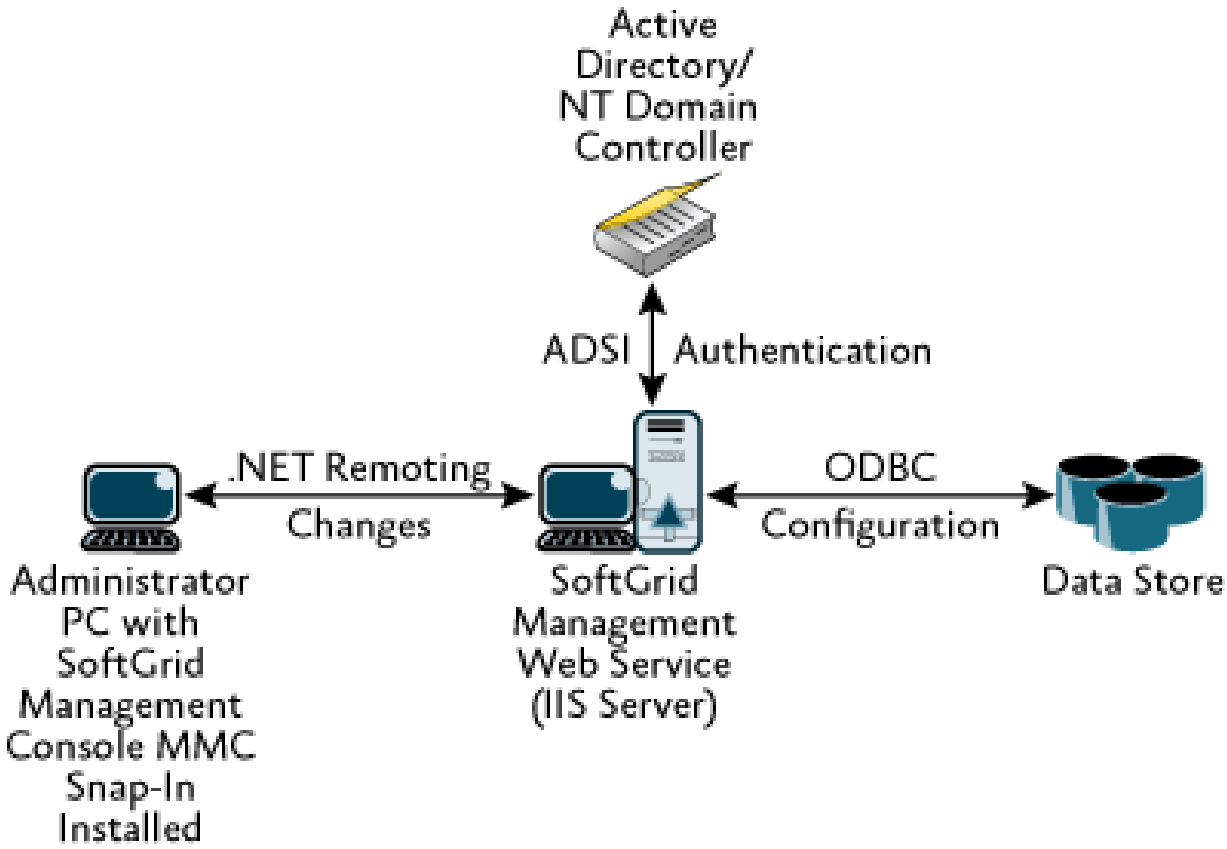


Рисунок 1.10 – Схема подключения тонких клиентов через систему SoftGrid

SoftGrid Client. Локально установленная служба, выполняемая на компьютере пользователя; взаимодействует с сервером виртуального приложения, чтобы получить потоковое приложение и кэшировать его для использования, даже если клиент не подключен к сети. Клиент также содержит среду SystemGuard для динамического выполнения виртуализованного приложения.

SoftGrid Sequencer. Инструмент на основе мастера, который используется для создания потокового приложения из исходного варианта программы. Этот процесс знаком пользователям таких программ, как WinINSTALL, которая создает .msi-файл на основе процедуры установки приложения. В сущности, преобразование приложения состоит из трех этапов: конфигурирование пакета; установка приложения, в ходе которого мастер отслеживает изменения, вносимые в операционную систему в ходе установки приложения, в том числе изменения в файловой системе, реестре и регистрации компонентов; собственно запуск приложения — важный этап, поскольку при первом использовании SoftGrid определяет приоритет данных для потоковой передачи клиенту в зависимости от того, какие компоненты используются первыми при запуске приложения (компонент, загружаемый в первую очередь, известен как Feature Block 1 — FB1 — потока). Поэтому нет необходимости ждать завершения пересылки перед запуском приложения.

Существуют дополнительные компоненты, интегрированные с решением System Center Configuration Manager (SCCM), которые позволяют использовать SCCM для дистрибуции, а также отдельный клиент для среды типа сервера терминалов (в том числе Citrix). В данной статье недостаточно места для описания этих компонентов, но о них следует знать, если предполагается использовать SCCM или серверы терминалов.

Что ждет виртуализацию в будущем? За последние несколько лет интерес к виртуализации вырос в разы, что можно увидеть по статистике Google Trends.

Тем не менее, в связи со сложностью и высокой стоимостью развертывания и поддержки виртуальной инфраструктуры, а также трудностью правильной оценки возвращения инвестиций, многие проекты по виртуализации увенчались неудачей. По результатам исследований, проведенных Computer Associates среди различных компаний, предпринявших попытки виртуализации, 44 процента не могут охарактеризовать результат как успешный. Это обстоятельство сдерживает многие компании, планирующие проекты по виртуализации. Проблему составляет также факт отсутствия по-настоящему грамотных специалистов в этой области.

2006 год стал для технологий виртуализации ключевым: множество новых игроков пришли на этот рынок, множество релизов платформ виртуализации и средств управления, а также немалое количество заключенных партнерских соглашений и альянсов, говорят о том, что в будущем технология окажется очень и очень востребованной. Рынок средств виртуализации находится в заключительной стадии своего формирования. Множество производителей аппаратного обеспечения заявили о поддержки технологий виртуализации, а это верный залог успеха любой новой технологии. Виртуализация становится ближе к людям: упрощаются интерфейсы для использования виртуальных машин, появляются, не закрепленные пока официально, соглашения об использовании различных средств и техник, упрощается миграция с одной виртуальной платформы на другую. Безусловно, виртуализация займет свою нишу в списке необходимых технологий и инструментальных средств при проектировании IT-инфраструктуры предприятий. Обычные пользователи также найдут свое применение виртуальным машинам. С ростом производительности аппаратных платформ настольных компьютеров появится возможность поддерживать на одной машине несколько пользовательских окружений и переключаться между ними.

Производители аппаратного обеспечения также не собираются оставаться на месте: помимо существующих техник аппаратной виртуализации, вскоре появятся аппаратные системы, нативно поддерживающие виртуализацию и предоставляющие удобные интерфейсы для разрабатываемого программного обеспечения. Это позволит быстро разрабатывать надежные и эффективные платформы виртуализации. Возможно, что любая устанавливаемая операционная система будет сразу

виртуализовываться, а специальное низкоуровневое ПО, при поддержке аппаратных функций, будет осуществлять переключение между запущенными операционными системами без ущерба для производительности.

Сама идея, заложенная в технологиях виртуализации, открывает широкие возможности по их использованию. Ведь, в конечном счете, все делается для удобства пользователя и упрощения использования привычных ему вещей. А можно ли на этом существенно экономить деньги, покажет время.

Виртуализация приложений — перспективное направление. С помощью SoftGrid несложно без значительных затрат преобразовать приложения в виртуальные. Кроме того, среда SystemGuard обеспечивает одновременное выполнение приложений, устраняя проблемы совместимости. Процесс экспериментальной виртуализации Microsoft Office XP и Office 2003 был гладким и интуитивно понятным. В следующей статье будет рассказано о применении виртуализованных приложений SoftGrid.

## 2 Центр обработки данных

### 2.1 Концепция

Суть нашей концепции предполагает, что в каждом университете или колледже должен существовать расширяемый и масштабируемый DATA CENTER – ЦОД – (Центр Обработки Данных). Занятия проходят в компьютерных классах, эти классы оборудованы тонкими или нулевыми клиентами. Все классы подключены к ЦОДу. (Рисунок 2.1). Для каждой дисциплины создаются специальные «образы» для проведения занятий по всем, изучаемым дисциплинам.

На переменах между занятиями оператор ЦОДа загружает необходимые «образы» для проведения занятия. Данные «образы» содержат только необходимое программное обеспечение и дополнительное программное обеспечение, которое отслеживает все действия студента. Все студенты получают однотипные задания. Отслеживающее программное обеспечение по окончании занятия создаст отчет о всех действиях студента, с помощью которого преподаватель может объективно оценить знания студента по данному предмету. Схема подключения тонких клиентов к Серверу предоставлена на Рисунке 2.1.

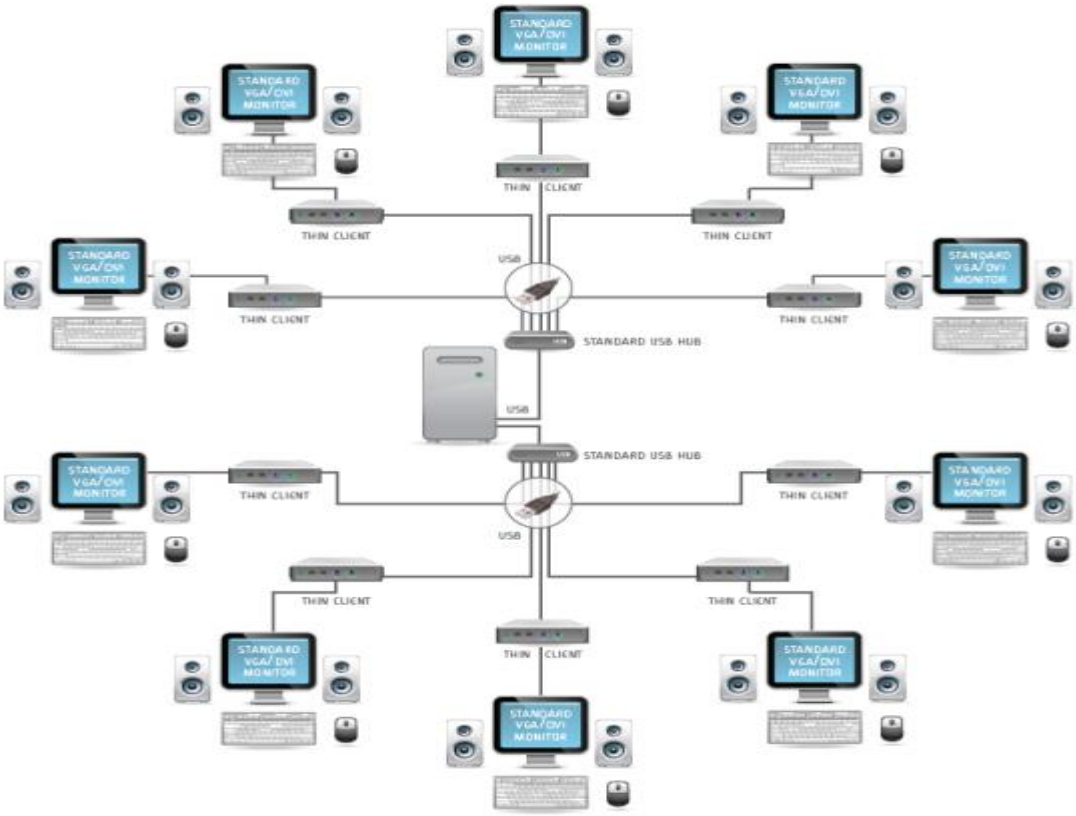


Рисунок 2.1 – Схема подключения тонких клиентов к Серверу

Программное обеспечение может анализировать действия студента на основе заранее заложенных критериев. После того как занятия будут завершены, следующий предмет можно проводить в том же кабинете, необходимо лишь загрузить новый чистый образ системы или приложения на тонкие клиенты.

Виртуализация приложений значительно упрощает процесс развертывания. Поскольку приложения больше не конфликтуют с версиями библиотек DLL или других общих ресурсов среды, необходимость тестирования новых приложений на наличие конфликтов с существующими приложениями сводится к минимуму. Виртуальные приложения могут работать рядом с обычными установленными приложениями, поэтому нет необходимости выполнять общую виртуализацию. Однако общая виртуализация помогает избежать многих проблем и сокращает время на обращение пользователей в службу поддержки. Эффективное решение виртуализации приложений позволяет управлять виртуальными и установленными приложениями с помощью общего интерфейса.

Виртуальные приложения SoftGrid. Виртуализация приложений позволяет запускать приложения локально и экономить ресурсы сервера, устраняет единые точки отказа и снимает ограничения на работу в сети. Решение для виртуализации приложений SoftGrid обеспечивает аналогичную «песочнице» среду под названием SystemGuard, в которой приложения могут выполняться на компьютерах пользователей без локальной установки операционной системы.

SystemGuard — виртуальная прикладная среда, которая содержит все необходимые приложению элементы, такие как файлы, данные реестра, COM-объекты и информация о среде. Совместимые с SoftGrid программы не требуется устанавливать в основной операционной системе, но они взаимодействуют с операционной системой в соответствии с определенными правилами, чтобы избежать излишнего дублирования данных в виртуализованном пространстве.

Как показано на Рисунок 2.2, приложение взаимодействует с ресурсами, которые представляются ей обычной операционной системой, с полными правами записи и чтения в среде SystemGuard. SystemGuard строго контролируемо взаимодействует с настоящей операционной системой. Информацию о настроечных параметрах можно прочитать, но не изменить. Данные профиля и документов могут быть изменены в операционной системе, которая позволяет сохранять данные и параметры среды между сеансами.

Рисунок 2.2. – Взаимодействие приложений в среде SystemGuard

Так же данное решение позволяет использовать виртуализацию приложений удаленно, все что необходимо это лишь VPN. (Рисунок 2.3)

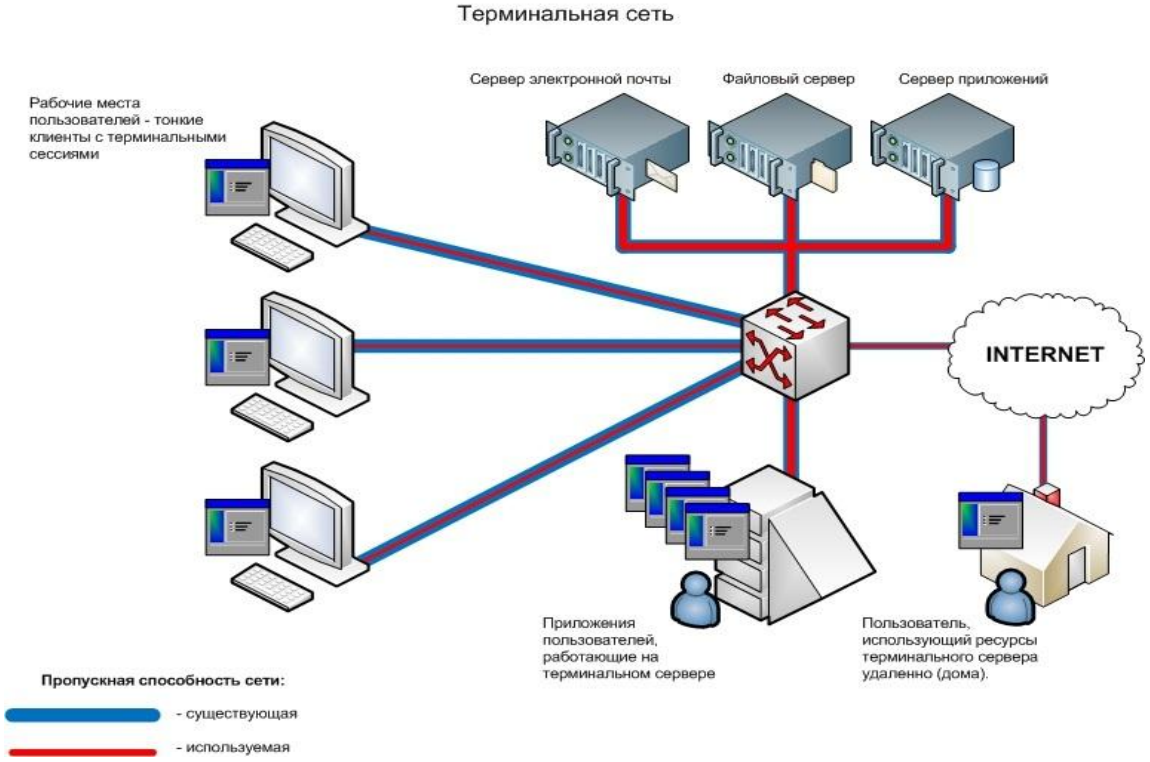


Рисунок 2.3 – Подключение через VPN

## 2.2 Основные преимущества концепции

– ЦОД является центром управления учебным процессом университета;

– обновление (upgrade) Hardware & Software проходит только на ЦОДе, что существенно экономит затраты на компьютерное оборудование;

– в компьютерных классах используются тонкие (thin) и нулевые (zero) клиенты;

– вычислительная мощность ЦОДа может использоваться для научных исследований;

– использование виртуализации увеличивает эффективность использования компьютерной техники с 10 % до70 - 80%;

– на кафедры ИС подготовлены по технологиям виртуализации и облачных вычислений три студента дипломника и один магистрант.

Организация учебного процесса. Занятия проходят в компьютерных классах которые будут оборудованы тонкими или нулевыми клиентами. Все классы подключены к ЦОДу. Для каждой дисциплины создаются специальные «образы» для проведения занятий по всем, изучаемым дисциплинам.

Согласно расписанию на переменах между занятиями оператор ЦОДа загружает необходимые «образы» для проведения занятия. Данные «образы» содержат только необходимое программное обеспечение для данной дисциплины. Студенты могут сразу приступать к выполнению заданий который поставил перед ними преподаватель. Эти задания будут расположены на локальном вебсервере. Каждый заходит под своим логин – паролем, и после этого студент выбирает необходимый предмет. (рисунок 2.4)

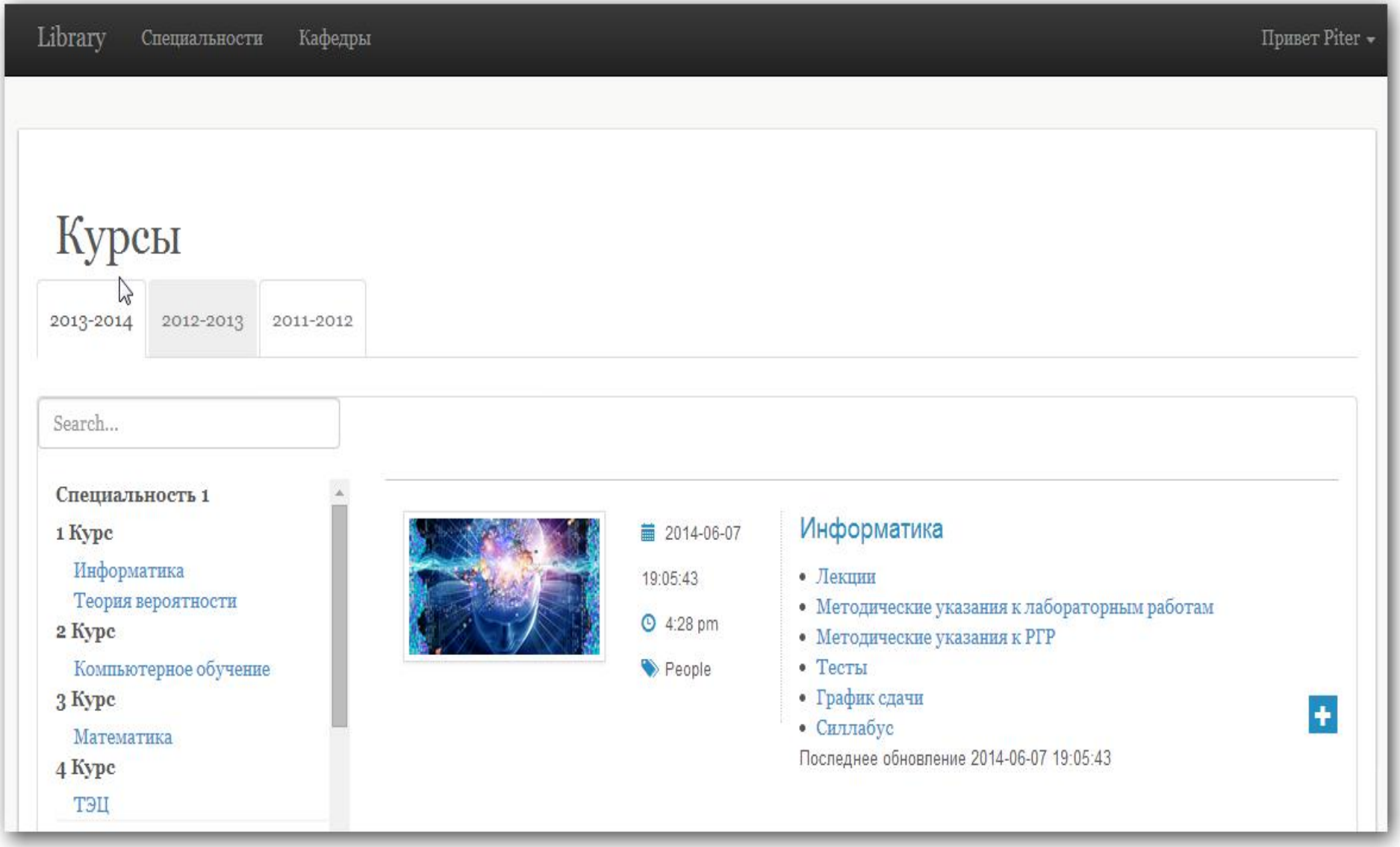


Рисунок 2.4 – Электронная библиотека

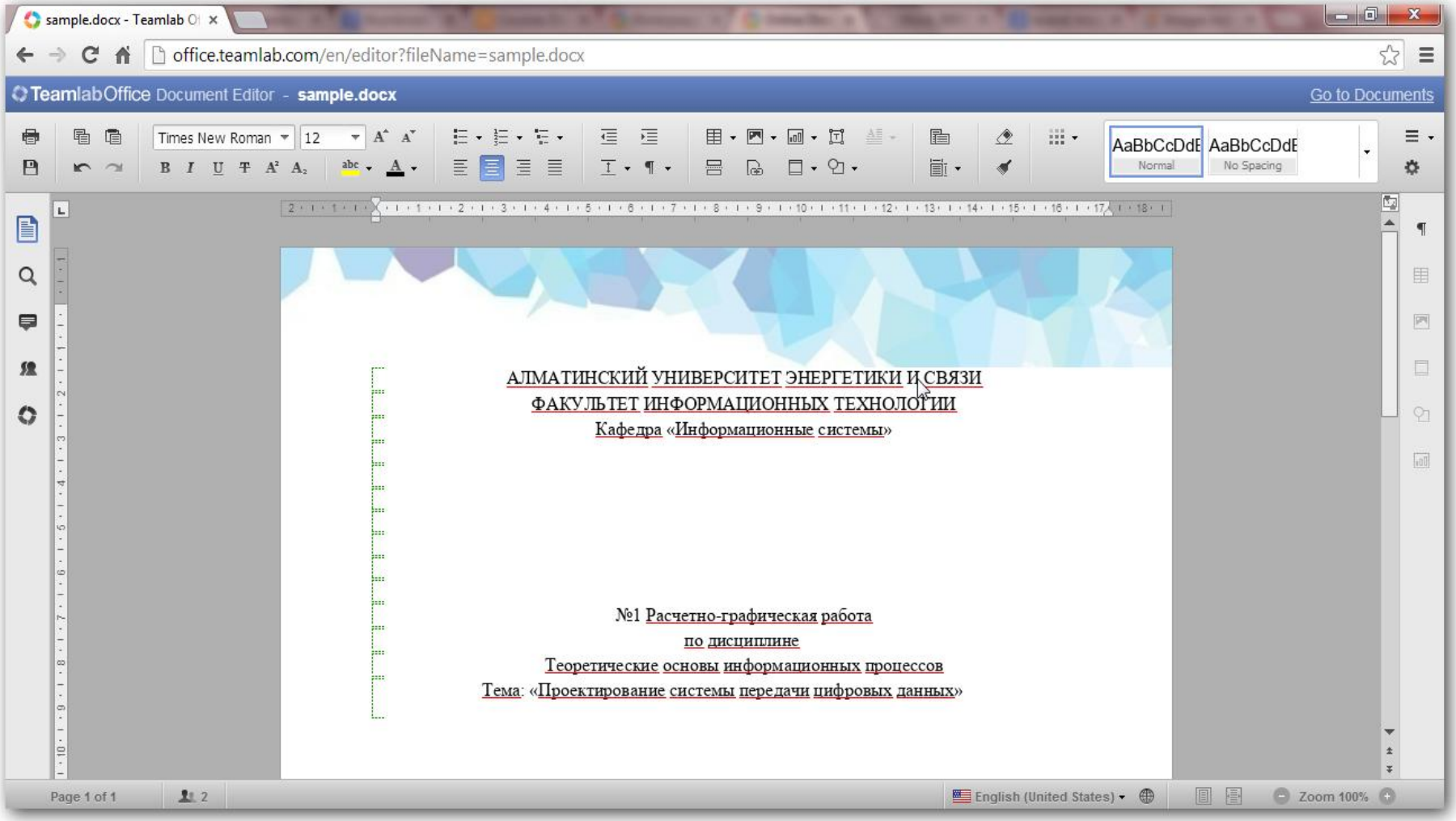


Рисунок 2.5 – Пример виртуализации приложения WORD

## 2.3 Защита данных

Наша концепция позволяет обеспечить надежность данных. Даже при успешном несанкционированном проникновении по сети на один виртуальный сервер другие виртуальные серверы остаются нетронутыми. Так же все данные о пользователе и его данных будут храниться на отдельном сервере, которому будет иметь доступ лишь сервер на котором будет виртуаилизрованно само приложение.

Так же значительно упрощается процесс делегирования полномочий.

Можно передать права на администрирование виртуальным сервером доверенному лицу, не переживая о пересечении полномочий при управлении процессами, размещенными на других серверах.

На одном компьютере возможно использование серверов под управлением различных операционных систем, например, Windows и FreeBSD.

Возможность поддержания старых операционных систем в целях обеспечения совместимости.

При выходе новой версии операционной системы, старую версию можно поддерживать на виртуальной машине, пока не будет полностью обкатана новая ОС. И наоборот, можно «поднять» новую ОС на виртуальной машине и опробовать ее без ущерба для основной системы.

Возможность изолировать потенциально опасные окружения.

Если какое-то приложение или компонент вызывает сомнения в его надежности и защищенности, можно использовать его на виртуальной машине без опасности повредить жизненно важные компоненты системы. Такую изолированную среду называют также «песочницей» (sandbox). Помимо этого, можно создавать виртуальные машины, ограниченные политиками безопасности (например, машина перестанет запускаться через две недели).

Возможность создания требуемых аппаратных конфигураций.

Иногда требуется использовать заданную аппаратную конфигурацию (процессорное время, количество выделяемой оперативной и дисковой памяти) при проверке работоспособности приложений в определенных условиях. Довольно сложно без виртуальной машины «загнать» физическую машину в такие условия. В виртуальных машинах – это пара кликов мыши.

## 2.4 Экономическая сторона

На первом месте идёт, как не странно, приобретение новых рабочих мест (и замена старых). В среднем рабочий ПК устаревает за 4-5 лет: «высыхают» блоки питания, «сыплются» накопители, очень часто (вы не представляете, насколько часто) выходят из строя мониторы. Рабочие места, отработавшие полный пятилетний цикл выглядят не лучшим образом: система «замусорена», винчестер регулярно похрустывает

ограничителями хода считывающих головок, конденсаторы в дешевом блоке питания приходят в некондиционное состояние, а уж про количество пыли внутри системника и говорить нечего.

Тонкий клиент (что All-in-One решения, что отдельные «коробочки») зачастую лишён всех этих проблем как конструктивно, так и идеологически, а сам по себе он не устаревает — вычислительные мощности консолидированы в серверной стойке, гибкость и масштабируемость которой заложена в самой идее виртуализированных рабочих мест.

Вторая расходная статья — электроэнергия. Расход электроэнергии даже с учётом потребления мощных серверов и СХД оказывается на 30-50% ниже, и это отношение тем больше, чем старше парк ваших компьютеров. Сказываются дешевые блоки питания с не самым высоким КПД, их деградация, нерациональное использование мощности.

Третья расходная статья —фонд оплаты труда. Допустим, у в кабинете 20 компьютеров и 2-3 сервера. Один системный/сетевой администратор обслуживает и “софтвейр”, и аппаратную часть., таких в кабинетов в одном корпусе может насчитываться до 20. Так же существуют другие корпусы которые могут находиться в нескольких километрах друг

от друга. Время от времени их надо чинить, чистить, обслуживать, а все эти операции требуют квалифицированных специалистов, и одним универсальным специалистом уже не обойтись. К тому же любая аппаратная поломка персонального компьютера приводит к простоям в обучении студентов. Внедрение тонких клиентов позволяет эффективно снизить как объём обслуживающего персонала, так и  затраты на ремонт и обслуживание тонких клиентов.

По мимо преимуществ которые были описаны выше, существуют и сложности, которые будут ждать при внедрении тонких клиентов

Затраты. Самые большие «потребители» финансов в VDI — тонкие клиенты, серверы и системы хранения данных. Причём затраты на покупку клиентов, серверов и жестких дисков примерно равны, а в сумме составляют более половины расходов.

Вторым по величине затратным пунктом является лицензионная сторона вопроса: серверные операционные системы, клиенсткие операционные системы, сам гипервизор и специализированное программное обеспечение требуют денег, и не малых, правда на обычные рабочие места точно также требуется купить операционку, весь набор софта и сказать сразу, что будет дешевле, обычные лицензии или лицензии для виртуализированных рабочих мест — сложно.

Третий пункт — обучение персонала. Затраты на обучения считаются весьма условными, так как для пользователей практически ничего не меняется, а администраторы и обслуживающий персонал обычно весьма широкопрофильные специалисты, да и технология виртуализации не должна создавать лишних проблем, для того её и разрабатывали, чтобы работать было проще и выгоднее.

Переход на виртуализированные рабочие места. Перед тем, как планировать полный переход учебного заведения на тонкие клиенты, следует убедиться в экономической эффективности перехода, рассмотреть основные модели перевода рабочих мест на VDI и оценить масштаб работ.

Необходимо расмотреть следующие вопросы:

Как давно обновлялся парк ПК?

- парк компьютеров обновлялся 1-2 года назад;
- парк компьютеров обновлялся 4-5 лет назад;
- парк компьютеров не обновлялся.

Сколько в учебномзаведении рабочих мест, которые быстро и безболезненно можно перевести на VDI?

- у нас много инженеров, работающих вместе над одним проектом (его разными частями);
- у нас много узкоспециализированных специалистов, которым требуются различные по своим возможностям ПК.

Насколько надежны текущие компьютеры и как часто требуют внимания?

– постоянно ломается;

– случаются поломки, но, как правило, всё лечится за несколько часов;

– у нас надёжные и качественные компьютеры, если проблемы и случаются, то только с софтом / по вине пользователя.

Если парк компьютеров обновлялся в течение последней пары лет, внедрять VDI сейчас — невыгодно, но можно начать готовить инфраструктуру: улучшить сеть, подготовить место для серверов, обновить СХД, если она есть.

Если компьютеры уже не молоды, но надёжны, их можно использовать как тонкие клиенты после соответствующей настройки и/или незначительной модернизации. Отработав ещё несколько лет, они позволят сэкономить на стартовой закупке тонких клиентов, а менять их можно по мере выхода из строя или просто через определённый промежуток времени, когда поддержка старого ПК станет менее выгодной, чем необслуживаемая простая железка с длительной enterprise — гарантией.

Если парк компьютеров изношен, а надёжность их вызывает у техперсонала опасения — менять такие компьютеры необходимо сразу же (или постепенно, по мере выхода из строя, переведя весь парк ПК в режим тонких клиентов).

Так как в учебных заведениях рабочие места используются для однообразных задач, внедрение тонких клиентов является отличной рекомендацией для таких предметов как : информатика, математический анализ, базы данных и многим другим предметам. Замена ПК на виртуализированные рабочие места позволит экономить на обслуживании парка техники и на счетах за электричество. Инженерным командам также подойдут виртуализированные рабочие места (в каких-то случаях VDI, в каких-то RDS, у обеих технологий есть свои особенности и достоинства), а мультимониторные тонкие клиенты позволят удобно работать даже с самым сложным софтом.

### 2.5 Окупаемость

Окупаемость напрямую будет зависеть от того, как и в каких количествах внедрются тонкие клиенты вместо обычных ПК. Использование текущих рабочих мест в качестве терминалов для VDI позволяет сэкономить на «стартовом пакете», но не избавляет от тех операционных расходов, которые связаны с обслуживанием этих компьютеров.

В данном случае требуется рассчитать, сколько будет стоить обслуживание рабочих мест, например, ещё 2-3 года (3 года — как раз тот срок, когда экономический эффект от внедрения VDI становится очевиден),

и насколько уменьшатся расходы за те же 2-3 года, если заменить эти компьютеры на тонкие клиенты.

Подробный анализ текущей обстановки в учебном заведнии, оценка предложений компаний-интеграторов VDI и высокая надёжность виртуализированных рабочих мест позволят снизить количество расходов через несколько лет, существенно улучшив как условия обучения студентов, так и качество работ преподавателей.

## 3 Разработка виртуальной библиотеки

Предваряя описание реально созданной в данной работе концепции тонких клиентов и оптимизации учебных процессов, кратко остановимся на общих соображениях [8,10], которые определили характер используемого подхода к внедрению разрабатываемых систем. Это обстоятельство является важным, в том числе с точки зрения новых подходов к креативному потреблению [11], а также к созданию новых организационных форм, таких как учебно-научные инновационные кластеры, создаваемые в настоящее время в Казахстане [12-14]. Кроме того, анализ, который дается ниже следуя [15], позволяет показать, что используемый в данной работе подход полностью отвечает представлениях о логистике инноваций, в частности, понятию инновационной лестницы [15].

Как отмечается в [8,10] любые изобретения существуют не сами по себе, они являются порождением существующего общего уровня развития науки и техники, поэтому те или иные связи между ними заведомо имеются, другое дело, что часто они носят опосредованный характер и выявить их удается далеко не всегда. Однако во многих случаях связи прослеживают вполне отчетливо, особенно если выйти за рамки устаревшей классификации отраслей знания, дисциплин и т.д.

Следовательно, «стартовую позицию», или же существующее положение дел в науке отдельной страны можно охарактеризовать с помощью логистико-инновационной карты.

Этот инструмент, в том числе, позволяет адекватно истолковать понятие инновационной связанности, которое во многом аналогично понятию геополитической связности [15]. А именно, случай, когда области на логистической карте тесно связаны друг с другом, говорит о том, что в данном государстве инновационные процессы носят системный характер, что отвечает высокой инновационной связности. Внедрение отдельного новшества не выпадет из общей картины, для этого не требуется экстраординарных усилий. И наоборот, если на рассматриваемой карте доминируют отдельные, не связанные друг с другом области, то инновационная связанность низка. Государство с низкой инновационной связностью, вкладывая средства в собственные научные исследования, фактически способствует решению научных и технических задач, поставленных другим государством [15]. В особенности это относится к исследованиям фундаментального характера. Действительно, если инновационная связность внутри страны низка, то нарабатываемые преимущества (в частности, связанные с опосредованным использованием результатов работ в области фундаментальных наук) не могут быть использованы в ее пределах. Однако, будучи опубликованными в открытой печати, или растиражированными в иной форме, они становятся «общим

достоянием». Принято считать, что достижения фундаментальной науки являются достоянием цивилизации в целом, однако на практике, указанными преимуществами пользуются государства, создавшие у себя системы, обладающие высокой логистико-инновационной связностью [15].

Таким образом, вопрос о повышении инновационной связности является ключевым для ускоренной модернизации стран СНГ. Поэтому поиск новшеств, которые могут стать ядром инновационного кластера, должен вестись целенаправленно: имеет смысл остановиться на вопросе о критериях выбора новшества, способного стать ядром инновационного кластера.

Общим критерием является возможность последовательного использования данной инновации как минимум в нескольких отраслях промышленности. Именно этот фактор определяет значение того или иного новшества с точки зрения повышения инновационной связности [15]. Кроме того, этот фактор определяет потенциал конкретной инновации с точки зрения работы на перспективу, что наиболее существенно для государств, осуществляющих переход от экономики, полностью ориентированной на добывающие отрасли, к наукоемкой.

Рассмотрим это обстоятельство подробнее. Инновации, как и любой продукт, формируют вполне определенный рынок, пусть и достаточно специфический. Поэтому задачи, возникающие при переходе к экономике, ориентированной на наукоемкие технологии, фактически сводятся к проникновению на международный рынок инноваций. Подчеркнем, что речь идет именно о проникновении на рынок, который во многом уже устоялся, где действует все та же жесткая конкуренция, характерная для любого другого рынка. При этом следует помнить, что, даже осуществляя те или иные мероприятия на своей собственной территории, государство, стремящееся осуществить переход к наукоемкой экономике, все равно тем самым стремится проникнуть на международный рынок инноваций и, следовательно, столкнется с жестким противодействием, принимающим самые разные формы [15].

Поэтому новшество, способное стать ядром инновационного кластера, должно быть в максимальной степени защищенным в части возможности удержания позиций на соответствующем сегменте международного рынка инноваций. При этом юридические методы защиты (получение патентов, лицензий и т.д.) в случае, когда затрагиваются серьезные интересы, очевидно, могут играть только вспомогательную роль. Защита по существу может быть только одна – действия на опережение. А именно, отдельно взятое новшество, не встроенное в систему, достаточно просто перехватить, или подавить, используя достаточно широкий арсенал методов ведения информационной войны в научно-технической сфере. Поэтому инновация обладает ценностью не только и не столько сама по себе, сколько как основа для внедрения последующих новшеств, для дальнейшего развития. Этот тезис, очевидно, находится в полном

соответствии с представлениями об общем системном подходе к инновациям.

Таким образом, представления об инновационной связности являются отнюдь не умозрительной конструкцией, а инструментом, позволяющим оценивать перспективы внедрения тех или иных новшеств. Несколько упрощая можно сказать так. Новшество, обладающее большим потенциалом с точки зрения представлений об инновационной связности, будет «подтягивать» также и внедрение наукоемких технологий в смежных отраслях техники, промышленности и т.д. Это и есть тот «локомотив», о котором много и долго говорят на самых различных уровнях.

Существует еще один критерий для оценки новшества с точки зрения его способности стать ядром инновационного кластера. Это – возможность поэтапного внедрения системы связаных друг с другом изобретений, технологий и т.д., когда каждый последующий шаг отвечает, не просто выполнению отдельного этапа внедренческих работ, но сопровождается созданием мощностей, способных приносить непосредственную прибыль (инновационная лестница). Представляется целесообразным оценивать существо результатов, получаемых в рамках программ фундаментальных исследований именно с указанной точки зрения.

В этом ключе в данной работе вначале предполагается внедрение систем на основе виртуализации рабочих мест, которые допускают дальнейший безболезненный переход к использованию виртуализации приложений, рассмотренных в главе 2.

Для оптимизации учебных процессов в высшем учебном заведении была разработана программа, позволяющая собрать все методички, лекции, дополнительные материалы на одном веб портале, с которого студенты и преподаватели смогут получить доступ к учебным материалам.

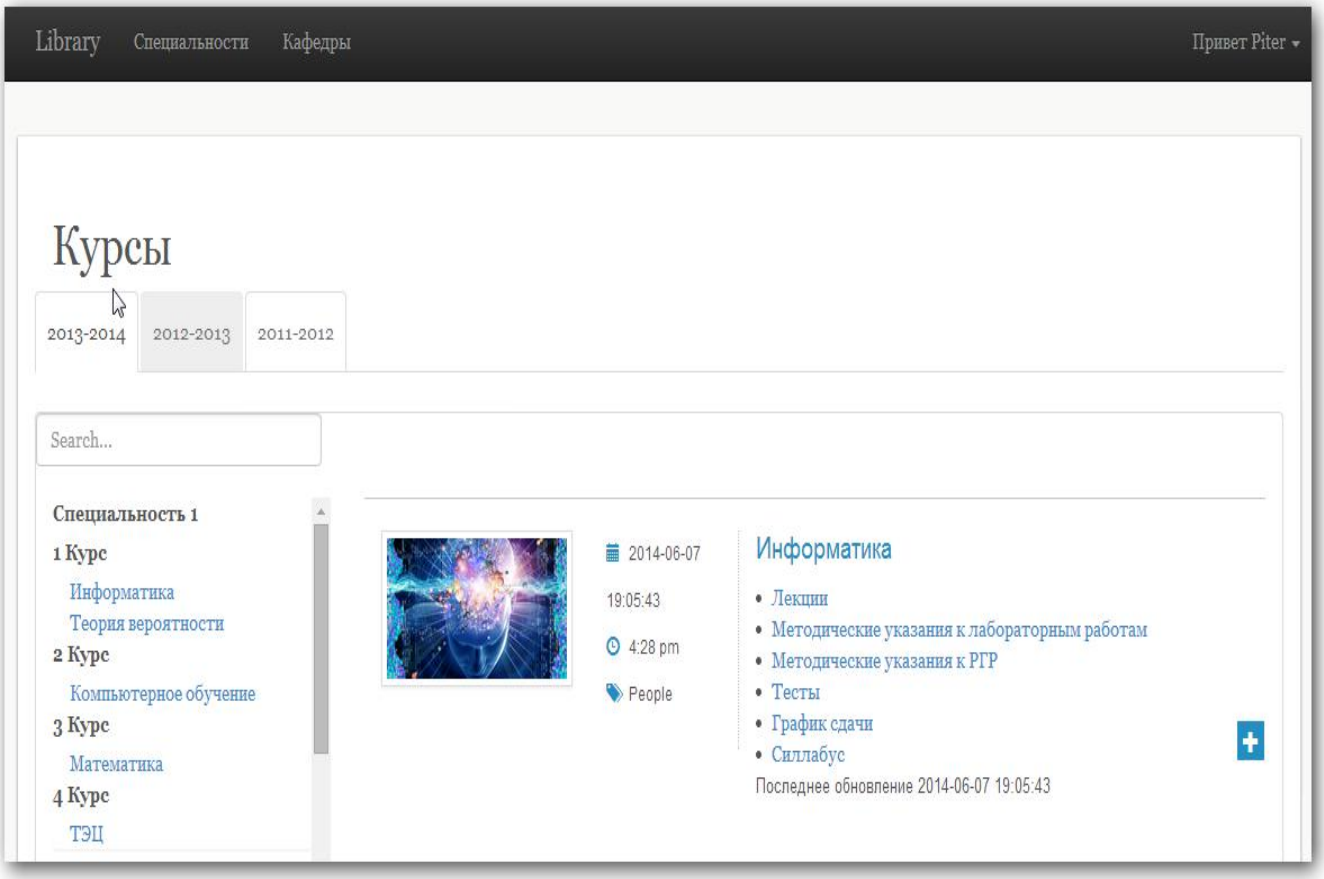


Рисунок 3.1 – Пример страницы Курсы в электронной библиотеке

Система была разработана на фреймворке Laravel 4.0. Платформа была выбрана ввиду широких возможностей данного фреймворка. Ключевые особенности, лежащие в основе архитектуры Laravel:

Пакеты (англ. packages) - позволяют создавать и подключать модули в формате Composer к приложению на Laravel. Многие дополнительные возможности уже доступны в виде таких модулей.

Eloquent ORM - реализация шаблона проектирования ActiveRecord на PHP. Позволяет строго определить отношения между объектами базы данных. Стандартный для Laravel построитель запросов Fluent поддерживается ядром Eloquent.

Логика приложения - часть разрабатываемого приложения, объявленная либо при помощи контроллеров, либо маршрутов (функций-замыканий). Синтаксис объявлений похож на синтаксис, используемый в каркасе Sinatra.

Обратная маршрутизация связывает между собой генерируемые приложением ссылки и маршруты, позволяя изменять последние с автоматическим обновлением связанных ссылок. При создании ссылок с помощью именованных маршрутов Laravel автоматически генерирует конечные URL.

REST-контроллеры - дополнительный слой для разделения логики обработки GET- и POST-запросов HTTP.

Автозагрузка классов - механизм автоматической загрузки классов PHP без необходимости подключать файлы их определий в include. Загрузка по требованию предотвращает загрузку ненужных компонентов; загружаются только те из них, которые действительно используются.

Составители представлений (англ. view composers) - блоки кода, которые выполняются при генерации представления (шаблона).

Инверсия управления (англ. Inversion of Control) - позволяет получать экземпляры объектов по принципу обратного управления. Также может использоваться для создания и получения объектов-одиночек (англ. singleton).

Миграции - система управления версиями для баз данных. Позволяет связывать изменения в коде приложения с изменениями, которые требуется внести в структуру БД, что упрощает развёртывание и обновление приложения.

Модульное тестирование (юнит-тесты) - играет очень большую роль в Laravel, который сам по себе содержит большое число тестов для предотвращения регрессий (ошибок вследствие обновления кода или исправления других ошибок).

Страничный вывод (англ. pagination) - упрощает генерацию страниц, заменяя различные способы решения этой задачи единым механизмом, встроенным в Laravel.

### 3.1 Организационная структура

Студенты могут заходить на сайт и скачивать необходимую информацию. Преподаватели загружают контент для учебы.

Описание операций:

Преподаватель осуществляет несколько видов операций:

– заполнение дисциплины учебными материалами.

Администратор осуществляет несколько видов операций:

– заведение Преподавателей и Зав.кафедры в систему;
– создание в начале года дисциплины.

Зав.кафедры может делать все тоже самое, что и Преподаватель, но так же может просматривать отчет за семестр, в котором отображены все дисциплины его кафедры и наличие учебных материалов

### 3.2 Перечень и описание основных сущностей предметной области

Факультет - учебно-научное и административное структурное подразделение высшего учебного заведения, осуществляющее подготовку студентов и аспирантов по одной или нескольким родственным специальностям, повышение квалификации специалистов, а также руководство научно-исследовательской деятельностью кафедр, которые он объединяет.

Дисциплина - курс, по которому в соответствии с программой и учебным планом ведется подготовка учащихся в рамках профиля учебного заведения и избранной ими специальности.

Преподаватель – Сотрудник, который занимается преподаванием Дисциплины.

Студент - учащийся высшего учебного заведения.

Зав Кафедры - руководитель кафедры в высших учебных заведениях, как правило, профессор, доктор наук. Заведующий кафедрой несёт персональную ответственность за организацию учебной, методической, научной (в том числе подготовка аспирантов) и иных работ на кафедре.

Электронная библиотека – упорядоченная коллекция разнородных электронных документов (в том числе книг), снабженных средствами навигации и поиска. Может быть веб-сайтом, где постепенно накапливаются различные тексты (чаще литературные, но также и любые другие, вплоть до компьютерных программ) и медиафайлы, каждый из которых самодостаточен и в любой момент может быть востребован читателем

### 3.3 Концептуальное проектирование

Для электронной библиотеки мы выявили следующие источники информации:

- преподаватели;
- зав.кафедры;
- администратор.

Сотрудник (который и является администратором) управляет системой, вводит данные в БД, выполняет запросы, обновляет БД.

- система должна быть проста в использовании. Интерфейс основного меню должен быть очень удобным;
- система должна обеспечивать безопасность. Это осуществляется с помощью паролей и прав доступа;
- система должна быть полностью интегрирована, т. е. должны быть исключены избыточность данных и избыточные обновления. Система должна гарантировать целостность БД.

Учитывая особенности данной системы, расмотренной в анализе предметной области, выделим основные таблицы и связи между ними, предоставленные на предварительной диаграмме сущностей, которая приведена на рисунке 3.2.

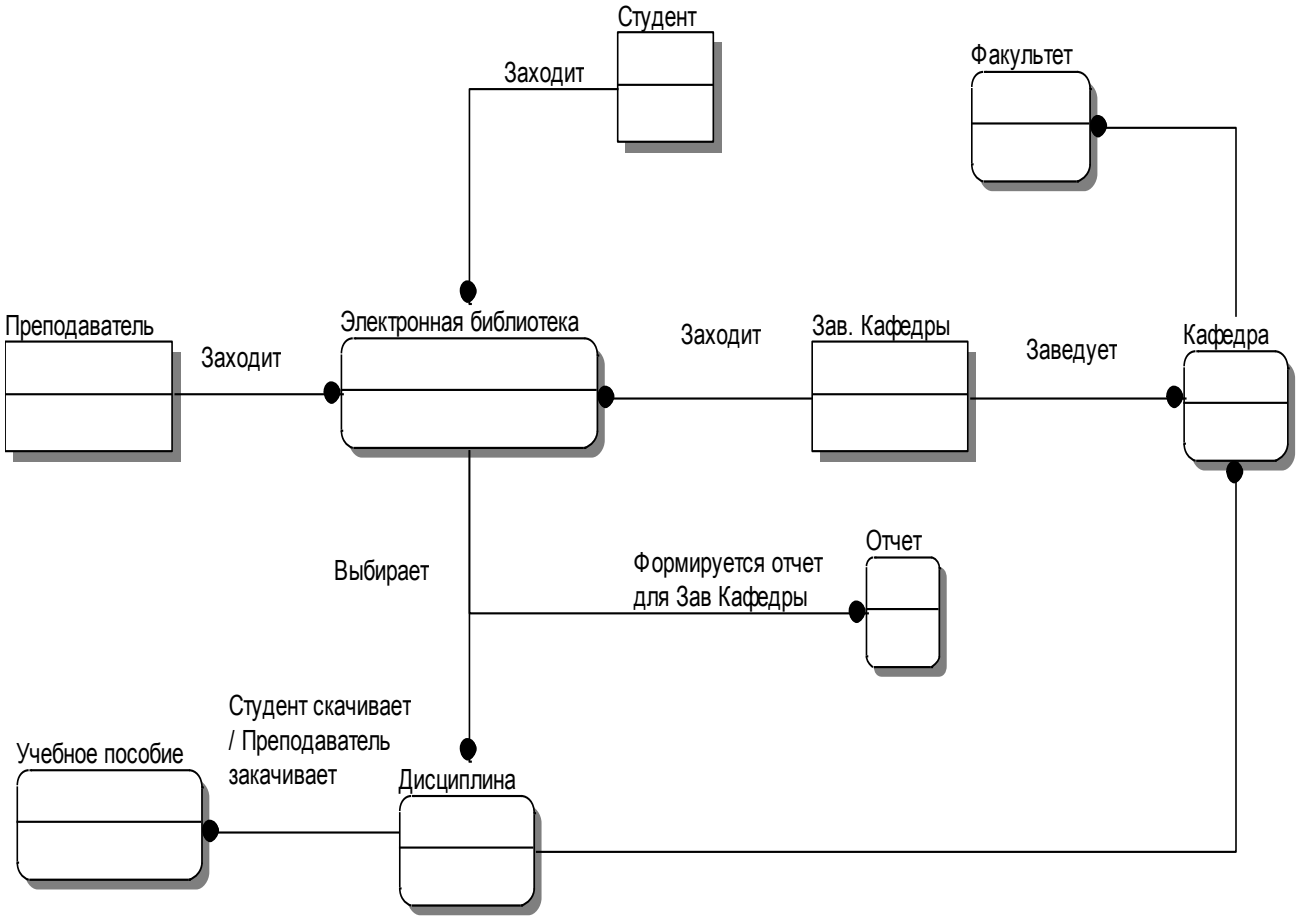


Рисунок 3.2 – Диаграмма сущностей

Диаграмма прецедентов. Для характеристики взаимодействия пользователей с базами данных применяется диаграмма прецедентов или вариантов использования:

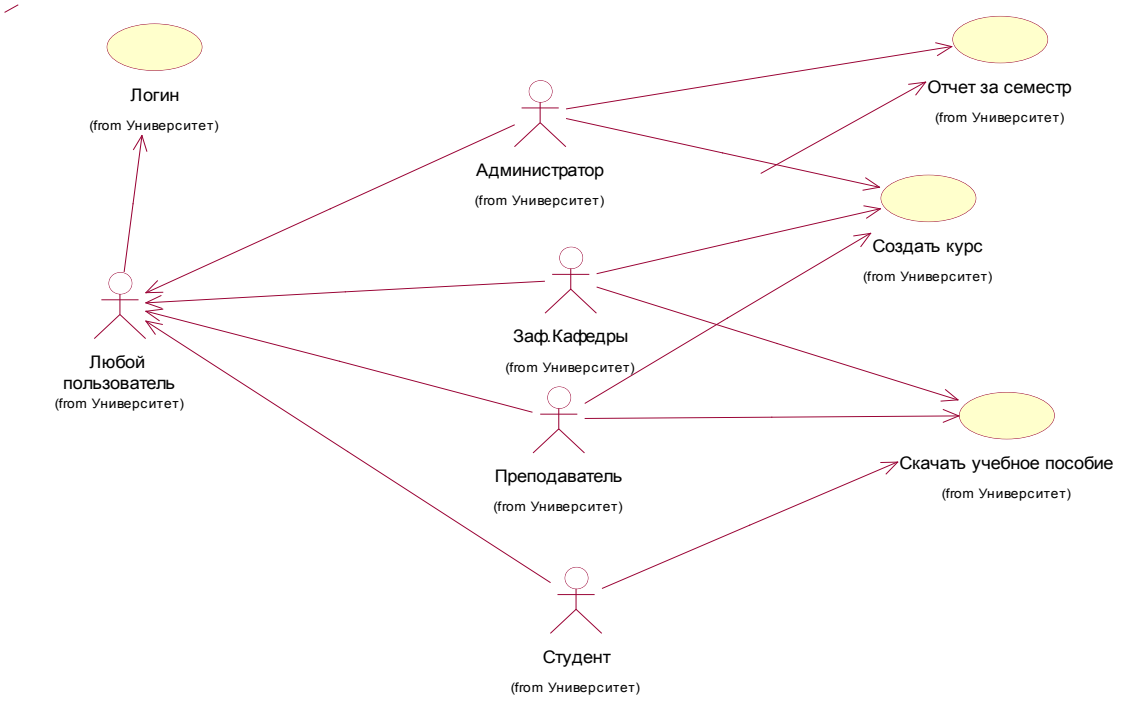


Рисунок 3.3 – Диаграмма прецедентов

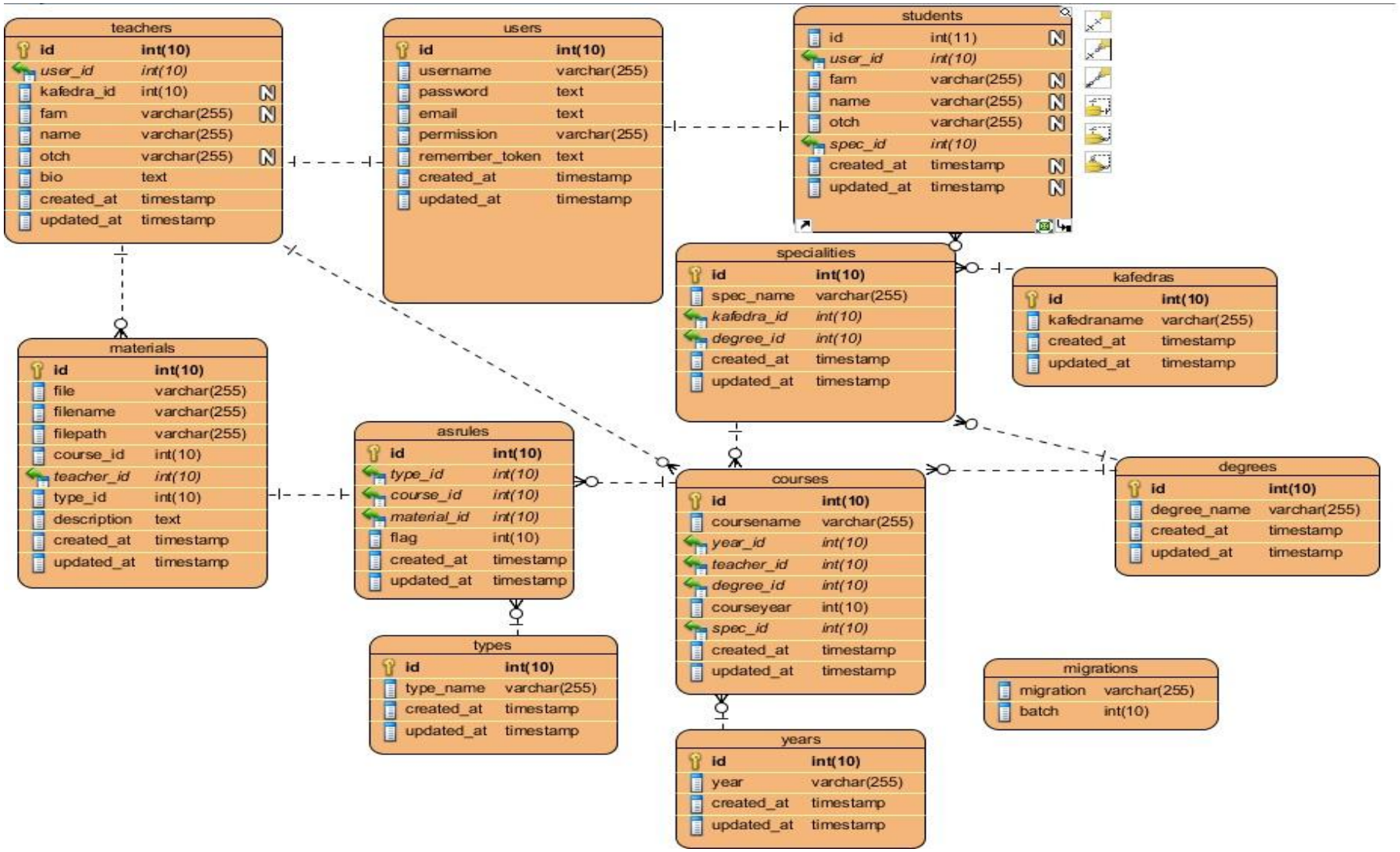


Рисунок 3.4 – ER диаграмма

Library Специальности Авторы Курсы Кафедры Привет Admin

2013-2014 2012-2013 2011-2012

| Преподаватель | Имя курса | Лекции | Силлабус | График сдачи | Карта | Методические указания к лабораторным работам | Методические указания к РГР | СРС | Тесты | Экзаменационные вопросы | Билеты | Паспорт | РП |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Нусипбеков Р.Е | Информатика | | | | | | | | | | | | |
| Нусипбеков Р.Е | Информатика для БВТ | | | | | | | | | | | | |
| admin A. | Математика | + | | | + | | + | | | | | | |
| admin A. | Базы данных | | | | | | | | | | | | |
| admin A. | Oracle | | | | | | | | | | | | |
| Mary . | ТЭЦ | | | | | | | | | | | | |
| Mary . | ТЭЦ 2 | | | | | | | | | | | | |
| Mary . | ЯиТП | + | | | + | | | | | | | | |
| Паркер П. | Информатика | | | | | | | | | | | | |
| Паркер П. | Английский язык | | | | | | | | | | | | |

Добавить Курс

Рисунок 3.5 – Пример страницы из электронной библиотеки - Курсы для заф.кафедры

## Заключение

Таким образом, в материалах данной работы показано, что внедрение тонких клиентов и разработка информационной системы для библиотеки, в которых используется виртуализация приложения, позволяет реализовать концепцию инновационной лестницы, понимаемой в духе работ крупнейшего казахского ученого проф. Г.А.Муна и его школы. (В данных работах показано, что именно реализация представлений об инновационных кластерах и инновационных лестницах может служить едва ли не единственно возможным средством прорывного развития РК в науке и технике.) По этой причине значение конкретного примера реализации инновационной лестницы, описанного в данной работе, не ограничивается техническими аспектами, что и определяет дополнительную важность полученных результатов.

А именно, в работе показано, что существует возможность последовательно наращивать наукоемкость и потребительскую привлекательность тонких клиентов, постепенно переходя от использования виртуализации операционных систем до виртуализации приложений.

В работе проведено детальное исследование целого ряда систем виртуализации, в котором были использованы, в том числе, наиболее передовые методы анализа.

Полученные результаты позволили определить оптимальную концепцию, обеспечивающего автоматизацию и оптимизацию учебных процессов.

Одним из основных достижений работы является также завершение работ над облачной библиотеке, которая уже внедрена в АУЭС, построенной на использовании платформа Laravel, по методу MVC, позволяющее безболезненное добавление функциий к существующей информационной системе.

В данный момент существуют множество Тонких клиентов, но не один из них не предлагает готовое решение для университетов, который позволял бы облегчить труд преподавателя в организации учебного процесса и оценивании знании студентов. Сама идея, заложенная в технологиях виртуализации, открывает широкие возможности по их использованию. Ведь, в конечном счете, все делается для удобства пользователя и упрощения использования привычных ему вещей. Тем самым мы решаем 2 проблемы

а) оптимизация процесса обучения и оценки знаний студента;

б) экономия средств университета затрачиваемых на модернизацию и поддержку существующей ИТ – инфраструктуры.

## Список литературы

1. Бережной А. Строим сетевую инфраструктуру на основе VMware Server// Системный администратор, 2007, №3, С. 14-18.

2. http://www.virtualbox.org – официальный сайт VirtualBox;

3. http://www.vmware.com – официальный сайт VMware.

4. Blade Servers and Virtualization: Transforming Enterprise Computing While Cutting Costs – by Barb Goldworm, Anne Skamarock, Wiley (February 12, 2007)

5.Оти, М. Технология виртуализации [Текст] / Майкл Оти // Windows IT Pro. - 2007. - №1. - С. 88-91

6. VMware Infrastructure 3: Advanced Operations Guide by Scott Herold, Ron Oglesby, Mike Laverick , the Brine Madden Company; second edition (August 1, 2008)

7. Виртуализация и Microsoft Virtual Server 2005 – Дитнер Роджер; Бином (2005)

8. Yergozhin Ye.Ye., Aryn Ye.M., Suleimenov I.E., Mun G.A., Belenko N.M., Gabrielyan O.A., Park N.T., Negim El-S. M. El-Ash., Suleymenova K.I. Nanotechnology versus the global crisis Seoul, Hollym Corporation Publishers, - 2010, 300 p.

9. Virtualization: From the Desktop to the Enterprise – by Chris Wolf, Erick M. Halter; Apress, 1 edition (May 25, 2005)

10. Сулейменова К.И., Мун Г.А., Сулейменова Р.Н., Пак И.Т. Инновационное развитие Казахстана: переход к постиндустриальной модели высшей школы // Известия научно-технического общества КАХАК., спец. Выпуск (39), 2012 г., с. 106-110.

11. Сулейменова К. И., Обухова П.В., Жакаев А., Гармашова Ю.М., Сулейменов И. Э., Концепция креативного потребления как одна из компонент становления «зеленой» энергетики // Тезисы международной конференции «В.И. Вернадский и глобальные проблемы современной цивилизации», - Симферополь, апрель 2013, с 164.

12. Пак И.Т., Мун Г. А., Бегисинова А.Б., Шалтыкова Д.Б., Схемы построения веб-ресурсов учебно–научного инновационного кластера // Тезисы международной конференции «В.И. Вернадский и глобальные проблемы современной цивилизации», - Симферополь, апрель 2013, с 207.

13. Мун Г.А., Сулейменов И.Э. Развитие нанотехнологии в Казахстане на современном этапе // Вестник национальной инженерной академии РК, 2011, №3 (41), С.108-112.

14. Мун Г.А., Сулейменов И.Э., Зезин А.Б., Абилов Ж.А., Джумадилов Т.К., Измайлов А.М., Хуторянский В.В. Комплексообразование с участием полиэлектролитов: Теория и перспективы использования в наноэлектронике (монография) / Библиотека

нанотехнологии. Выпуск 2. Алматы – Москва-Торонто – Рединг: Изд-во LEM, 2009, 256 с.

15. Ергожин Е.Е., Арын Е.М., Сулейменов И.Э., Беленко Н.М., Габриелян О.А., Сулейменова К.И., Мун Г.А. Нанотехнология. Экономика. Геополитика. / Библиотека нанотехнологии. Алматы – Москва-София-Антиполис – Симферополь: Изд-во ТОО «Print-S», 2010, 227 с.

16. Синицын В. VirtualBox – виртуализация под GPL//Системный администратор, 2007, №3, С. 20-23.

17.Самойленко, А. Виртуализация: новый подход к построению IT-инфраструктуры - 2007. - (http://www.ixbt.com/cm/virtualization.shtml)

## Определение

Thin client – Тонкий клиент – компьютер с минимальным программным обеспечением.

Zero client – Нулевой клиент – компьютер без программного обеспечения.

Gold image – Золотой образ – комплект программного обеспечения для установки операционных систем.