

Министерство образования Республики Беларусь  
Учреждение образования  
«Гомельский государственный университет  
имени Франциска Скорины»

## **АКТУАЛЬНЫЕ ВОПРОСЫ ФИЗИКИ И ТЕХНИКИ**

Материалы IV Республиканской  
научной конференции  
студентов, магистрантов и аспирантов

(Гомель, 15 апреля 2015 года)

В трёх частях

Часть 3

Гомель  
ГГУ им. Ф. Скорины  
2015

УДК 530

В материалах республиканской конференции рассматриваются актуальные общетеоретические и прикладные проблемы физики и техники.

Издание состоит из трёх частей. В третьей части опубликованы доклады, в которых обсуждаются вопросы моделирования физических процессов и автоматизации в образовании и компьютерных технологиях.

Статьи участников конференции размещены в алфавитном порядке по отдельным секциям.

Издание предназначено для студентов, учителей школ, гимназий, колледжей, преподавателей вузов.

**Редакционная коллегия:**

А. В. Рогачев (главный редактор),  
О. М. Демиденко, И. В. Семченко, С. А. Хахомов,  
А. Н. Сердюков, Ю. В. Никитюк,  
А. Н. Купо (ответственный секретарь),  
В. В. Андреев, Е. А. Дей, В. Д. Левчук, А. В. Воруев,  
В. Н. Мышковец, В. Е. Гайшун, Е. Б. Шершнев, Т. П. Желонкина

**ISBN 978-985-577-047-4 (Ч. 3)** © Учреждение образования  
**ISBN 978-985-577-050-4** «Гомельский государственный  
университет имени Франциска  
Скорины», 2015

## **Секция 3 «Автоматизация исследований»**

### **Председатели:**

Левчук Виктор Дмитриевич, канд. техн. наук, доцент,

Ворув Андрей Валерьевич, канд. техн. наук, доцент

**Н.В. Корзубова (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Леванцов**, ст. преподаватель

### **ПЛАТФОРМА «1С:ПРЕДПРИЯТИЕ» КАК СРЕДСТВО РАЗРАБОТКИ**

Термин «1С:Предприятие» обозначает систему программного обеспечения, в которую входят и платформа, и наборы прикладных решений (разного масштаба и разной отраслевой специфики), а также различных методик. Фактически платформа и есть средство разработки, но работает она как на этапе создания программ, так и при их выполнении.

«1С:Предприятие» – не универсальное, а специализированное средство разработки. Ни один инструмент не может претендовать на решение всех мыслимых проблем, однако универсальные средства разработки ориентированы на достаточно широкий спектр задач, а платформа «1С» – на задачи автоматизации бизнеса.

В платформе «1С:Предприятие» механизм бизнес-процессов полностью интегрирован в систему таким образом, что ни разработчик ни пользователь не видят «швов» разделяющих этот механизм и другую функциональность. Этот механизм включает средства для описания в прикладном решении схем бизнес-процессов, и их маршрутизации, для формирования заданий, выполняющихся в каждой точке маршрута, для управления бизнес-процессом и организации его связи с другими функциями прикладного решения.

Бизнес-процессы формулируют и реализуют правила выполнения отдельных операций и их взаимосвязь, что позволяет значительно сократить или даже полностью исключить ошибки в ходе выполнения бизнес-процесса, связанные с человеческим фактором. Работа с простым списком заданий позволяет сотрудникам сконцентрироваться на выполнении своих непосредственных обязанностей.

Использование механизма бизнес-процессов позволяет формализовать организационную деятельность и возложить обязанности по управлению совместной работой сотрудников на плечи прикладного

решения, что ведет к более эффективному использованию рабочего времени.

«1С:Предприятие» как предметно-ориентированная среда разработки имеет определенные преимущества. Поскольку круг задач более точно очерчен, то и набор средств и технологий можно подобрать с большей определенностью.

Платформа «1С:Предприятие» содержит такие инструменты для выполнения поставленных задач, как визуальное описание структур данных, написание программного кода, визуальное описание запросов, визуальное описание интерфейса, описание отчетов, отладка программного кода, профилирование. В ее составе развитая справочная система, механизм ролевой настройки прав, инструменты создания дистрибутивов, удаленного обновления приложений, сравнения и объединения приложений, ведения журналов и диагностики работы приложения, создания Web-приложений и приложений, а также поддержка коллективной разработки, версионирования и прочее.

Разработка в «1С:Предприятие» строится на основе общей модели работы приложения, предлагаемой платформой «в обязательном порядке», т. е. основные и наиболее сложные архитектурно-технологические решения предлагаются разработчикам в готовом виде.

В «1С:Предприятие» процесс написания программного кода – не центральный элемент разработки программного обеспечения. Приложение разрабатывается, прежде всего, как структура метаданных. Код пишется в определенных узлах приложения «по необходимости», чтобы переопределить, если это нужно, стандартное поведение или написать ту часть бизнес-логики, которая требует именно алгоритмической формулировки. Идея разработки на основе метаданных начинает активно использоваться и в универсальных системах, но в предметно-ориентированной среде разработки она дает существенно больший эффект, так как структура метаданных четко ориентирована на круг решаемых системой задач.

Принятое в «1С:Предприятие» построение основной части приложения на основе стандартных прототипов прикладных объектов. Эта модель отличается от классического подхода (объектно-ориентированного программирования и работы с таблицами базы данных или отображаемыми в базу данных произвольными сущностями). Фактически система предоставляет не один базовый класс для построения прикладных объектов приложения, а несколько, каждый из которых имеет специализированную функциональность и предназначен для отображения в приложении объектов предметной области, обладающих схожими свойствами и ролью в бизнес-логике. Разработчик

использует эти прототипы для создания объектов приложения, которые уже являются финальными.

Прототипы применяются с некоторой параметризацией, определяющей необходимые в конкретном случае свойства и особенности поведения. Такой подход фактически обеспечивает построение приложения на основе определенной прикладной модели, в которой каждый объект играет определенную роль, и система хорошо знает эту роль, что позволяет ей автоматически выполнять существенную часть операций. Такой подход позволяет использовать в платформе новые архитектурные и технологические решения без смены прикладной модели и без изменения приложения.

Еще одна особенность «1С:Предприятие» как предметно-ориентированной среды разработки – особое отношение к подбору технологических возможностей, предоставляемых разработчику. Причем высокая степень «готовности» включает и простоту освоения, и «гладкость» интеграции с общей функциональностью и другими технологическими возможностями системы. Фактически платформа позволяет разработчику прикладных решений задействовать необходимые и современные технологии своевременно, максимально просто и без радикальных изменений в своем приложении.

Построение системы на основе технологической модели работы приложения, метаданных и прикладной модели работы приложения позволяет существенно упростить и ускорить разработку. Опираясь на метаданные и единую модель, все компоненты системы взаимодействуют между собой без существенных усилий со стороны разработчика.

В версиях 7.x платформы «1С:Предприятие» использовался другой подход, касающийся применения технологий. Система была в основном замкнута. Подход предполагал использование строго ограниченного набора средств и технологий. Это в свою очередь существенно ограничивало и спектр применения системы – как по масштабированию решений, так и по широте областей применения.

При разработке и развитии платформы «1С:Предприятие 8.0» подход был изменен. Система не ограничивается в использовании современных технологий, но ориентируется на их максимально удобное и простое включение в общую инфраструктуру разработки приложения. Реализацию этого подхода можно наблюдать в постепенном включении в платформу целого спектра технологических решений. Это права доступа к данным на уровне записей, широкие возможности интернационализации, работа с различными Интернет-протоколами, кластер серверов, управление блокировками данных, мощная поддержка XML и т. д.

В платформе появились более сложные предметы, требующие хорошего понимания, и само количество предметов стало больше. Однако эта сложность не захлестывает разработчика приложения. Все механизмы можно использовать опционально и вводить по мере надобности. Разработчику, переходящему с предыдущей платформы, нужно понимать, что это вполне разумная плата за возможность использовать при быстрой разработке современные технологии и решать задачи, которые в версии 7.x не могли решаться в принципе.

Использование «1С:Предприятие» стоит рассматривать для решения тех задач, для которых оно предназначено, – автоматизации управления и учета. Также важным критерием является оценка затрат на разработку и сопровождение системы. При этом затраты вполне можно оценить количественно. Скорость разработки в «1С:Предприятие» обычно выше в 2–10 раз и стоимость соответственно в разы ниже.

**Н.В. Корзубова (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Леванцов**, ст. преподаватель

## **ТЕХНОЛОГИЧЕСКИЕ СРЕДСТВА РАЗРАБОТКИ**

Для описания специфических алгоритмов обработки информации и создания интерфейса, ориентированного на удобное представление описанных в конфигурации данных, в системе «1С: Предприятие» используются несколько технологических механизмов.

*Встроенный язык.* Прикладные решения в «1С:Предприятии» не программируются (кодируются) целиком, большая их часть описывается параметрически – в виде структур метаданных, с помощью дизайнера форм, отчетов и т. д.

*Встроенный редактор форм.* Он позволяет оформить большинство окон, которые используются в системе для ввода и просмотра предметной информации (формы документов, справочников, настройки отчетов).

*Встроенный редактор табличных документов.* Для всех выходных документов в системе предусмотрен единый формат – формат табличных документов. Он может быть использован как для создания небольших документов с очень сложной структурой линий, так и для объемных ведомостей, журналов и других подобных документов.

*Конструкторы* – вспомогательные инструменты, облегчающие разработку стандартных элементов системы «1С:Предприятие». В системе имеются, например, конструкторы форм констант, справочников,

документов, журналов документов, отчетов и других объектов, конструкторы печатных форм, конструкторы движений регистров и другие.

*Система настройки пользовательского интерфейса.* Для того чтобы интерфейс конкретной конфигурации системы полностью отражал настроенные структуры данных и алгоритмы, в системе «1С:Предприятие», помимо редактора диалоговых форм и табличных документов, предусмотрена возможность настройки командного интерфейса системы.

*Подсистемы.* Конфигуратор позволяет на этапе проектирования в рамках одной конфигурации выделить различные подсистемы. Для каждой подсистемы можно указать объекты конфигурации, которые в нее входят. Фактически подсистемы определяют основные разделы конфигурации, с которыми будет работать пользователь.

*Система настройки прав доступа (роли).* Данная система позволяет описывать наборы прав, соответствующие должностям пользователей или виду деятельности.

*Отладчик.* Для удобства разработки конфигурации в системе предусмотрен отладчик. Отладчик позволяет проследивать исполнение программных модулей конфигурации, замерять сравнительное время исполнения, просматривать содержимое переменных.

*Хранилище конфигураций.* Для групповой разработки конфигурации разработчики используют механизм хранилища конфигурации. Он позволяет распределить права доступа по модификации объекта конфигурации и производить необходимые изменения одновременно, а не последовательно.

*Поддержка конфигураций.* Для удобства проведения обновлений конфигураций предусмотрен механизм формирования разработчиками типовых конфигураций файлов поставки и комплектов поставки (включают программу установки), а также механизм обновления типовых конфигураций, находящихся на поддержке.

**А.А. Котович (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **Н.А. Шаповалова**, ст. преподаватель

## **ИННОВАЦИОННЫЙ СЕРВИС РЕАЛИЗАЦИИ АВТОЗАПЧАСТЕЙ, ПОСТРОЕННЫЙ НА ТЕХНОЛОГИИ WHITE LABEL**

Целью данного проекта ставится разработка агрегатора автомобильных запчастей, который представляет из себя инновационный продукт с внедрением технологии White Label, позволяющего максимально

удобно и быстро искать и приобретать запчасти и детали для автомобилей, а также предоставить предпринимателям возможность организовать собственный сервис реализации запчастей на базе нашего продукта. Данная система будет включать в себя не только web-сервис, но и реализацию для мобильных платформ. Планируется внедрение информационных баз от различных поставщиков для возможности подбора деталей лучшего качества и максимально выгодной цены. Так же создать базу СТО, услуги которых будут предлагаться для работы с выбранным автомобилем и установки или замены необходимых деталей, предлагаемых на основе пользовательских запросов.

Для реализации в жизнь данной цели нам необходимо разработать и внедрить следующие мероприятия:

- разработка концепции дизайна интерфейса;
- создание максимально удобного интерфейса пользователя;
- на начальном этапе, для привлечения аудитории необходимо загрузить базы данных по меньшей мере одного крупного поставщика;
- разработка функциональной части;
- проектирование фильтров.

На начальном этапе, для привлечения аудитории необходимо загрузить базы данных по меньшей мере одного крупного поставщика. В качестве поставщика на начальном этапе была выбрана база компании ArmTek – поставщика на рынке запасных частей для легковых и грузовых автомобилей иностранного производства. База была представлена в виде csv-файлов общим размером 16Gb. В качестве базы описаний соответствия была взята база компании Tecdoc, которая предоставляет собой web-каталог автозапчастей в соответствии с маркой и моделью автомобиля, а также разбиение их на категории. Входной информацией являются таблицы базы данных.

На основе исследования сервисов автомобильной тематики и интернет-магазинов были выявлены необходимыми следующие разделы / формы:

- фильтр конфигурации авто для составления выборки;
- навигация по категориям товаров (breadcrumbs);
- листинг товаров по запросу и фильтр спецификации выборки;
- карточка товара;
- корзина заказа и расчёта суммы;
- профиль пользователя;
- управление сервисом.

Фильтр конфигурации автомобиля, а для зарегистрированных пользователей – гараж с возможностью добавления нескольких автомобилей,



необходим для фильтрации автозапчастей и аксессуаров на конкретную модель автомобиля.

Навигация по категориям или хлебные крошки (breadcrumbs) используются для отображения вашего текущего положения на сайте, относительно главной страницы.

Фильтр спецификации выборки и листинг результатов по запросу и есть основная часть сайта, а именно это отображение товаров по выделенным критериям поиска. Все фильтры разрабатывались отдельно для каждой группы товаров. Примером служили фильтры портала exist.

Карточка товара представляет собой модальное окно, открывающееся при выборе конкретного товара и содержит описание товара, фотографии, отзывы и оценку.

Корзина заказа и расчёта итоговой суммы представляет собой список ваших желаемых к покупке товаров и подсчет их общей стоимости для продолжения заказа.

Профиль пользователя представляет собой карточку зарегистрированного пользователя с возможностью управления и изменения автомобилей в так называемом гараже, а также с возможностью просматривать завершённые сделки, список желаний и настройки портала.

В качестве движка сайта на первое время была выбрана бесплатная расширяемая CMS OpenCart, ориентированная на создание интернет-магазинов. Может быть установлена на любом веб-сервере с поддержкой PHP и MySQL. Является свободным программным обеспечением с GNU General Public License – другими словами, распространяется бесплатно. OpenCart построен по принципу MVC и может быть установлен на любом веб-сервере с поддержкой PHP и MySQL. Структура OpenCart очень простая. На базе OpenCart можно создать интернет-магазина любой сложности. Сообщество OpenCart состоит из более 46 000 участников по всему миру, благодаря которому существует более 9 000 бесплатных и коммерческих дополнений позволяющие изменять или дополнять функционал магазина на OpenCart самым разным образом.

С точки зрения программиста OpenCart удобен следующим:

- данная среда совместима с PHP 5.x и MySQL 4.x, 5.x;
- код соответствует основным принципам паттерна Model-View-Controller;
- совместимость со всеми основными браузерами;
- встроенная многоязычность;
- неограниченное число разделов и товаров;
- поддержка шаблонов, модулей и дополнений;

– поддержка «Osmod» (Virtual Quick Mod), что значительно упрощает разработку расширений и их перенос при обновлении версии OpenCart.

С точки зрения администрирования система выигрывает за счет следующих пунктов:

– OpenCart поддерживает неограниченное количество продуктов и разделов категорий;

– поддержка физических и виртуальных (загружаемых) товаров;

– легкость резервного копирования и восстановления данных;

– статистика товаров и заказчиков;

– многоязычная поддержка (русский язык по умолчанию);

– поддержка нескольких валют.

При этом, благодаря технологии White Label, которая дает возможность под собственным брендом оказывать услуги и продавать товары, фактически произведенные другой компанией, при выборе товара доступ всегда организовывается лишь к базе текущего партнера, при этом база поставщика всегда находится в актуальном состоянии и защищена от изменений, ведь в случае изменения актуальность версии теряется и база перезаписывается. Для пользователя разница заключается лишь в том, что даже являясь владельцем дочернего магазина доступа к содержимому базы он не имеет. Т. е. несмотря на действия защиты методом запрета, у пользователя остаётся широкий круг возможностей.

Основные задачи, которые решает данный проект – предоставление универсального и понятного интерфейса для организации ремонта и модификации автомобиля в так называемом режиме «на диване». Возможность выполнения максимально сложной и полной фильтрации по базам автомобилей, автозапчастей, поставщиков и СТО.

**К.В. Красотин ( «БТЭУ», Гомель)**

Науч. рук. **С.В. Карпенко**, канд. экон. наук, доцент

## **ПРОЕКТИРОВАНИЕ ЛОКАЛЬНОЙ СЕТИ ПРЕДПРИЯТИЯ «ГОМЕЛЬСКИЙ ЗАВОД ТОРГОВОГО ОБОРУДОВАНИЯ»**

Компьютерная сеть – это неотъемлемая часть современной деловой инфраструктуры.

Реализация предложенного проекта позволит сократить бумажный документооборот внутри организации, повысить производительность труда, сократить время на обработку информации с использованием специализированных приложений, хранящихся на сетевом сервере,

и работать с общими устройствами: принтерами, факсами и другой периферией. Таким образом, решится проблема окупаемости и рентабельности внедрения локальной сети.

За счет внедрения в организации данного проекта ЛВС и подключения к глобальной сети Internet, предприятие получит практически неограниченные информационные возможности, связь с другими структурными подразделениями и вышестоящими органами.

Существует большое количество технологий: Ethernet, FDDI, Token Ring, ARCNet, ATM, UltraNet и другие.

Технология Ethernet была разработана в 1970 г. Исследовательским центром в Пало-Альто, принадлежащем корпорации Херох. В 1980 г. на его основе появилась спецификация IEEE 802.3. Пожалуй, самой характерной чертой Ethernet является метод доступа к среде передачи – CSMA/CD (carrier-sense multiple access/collision detection) – множественный доступ с обнаружением несущей. Перед началом передачи данных сетевой адаптер Ethernet «прослушивает» сеть, чтобы удостовериться, что никто больше ее не использует. Если среда передачи в данный момент кем-то используется, адаптер задерживает передачу, если же нет, то начинает передавать. В том случае, когда два адаптера, предварительно прослушав сетевой трафик и обнаружив «тишину», начинают передачу одновременно, происходит коллизия. При обнаружении адаптером коллизии обе передачи прерываются, и адаптеры повторяют передачу спустя некоторое случайное время (естественно, предварительно опять прослушав канал на предмет занятости). Для приема информации адаптер должен принимать все пакеты в сети, чтобы определить, не он ли является адресатом.

Различные реализации – Ethernet, Fast Ethernet, Gigabit Ethernet – обеспечивают пропускную способность соответственно 10, 100 и 1000 Мбит/с.

При конфигурировании сети Ethernet между конечными компьютерами разрешается использовать не более 4 концентраторов, 5 отрезков кабелей и 3-х нагруженных сегментов. Нагруженным сегментом называется концентратор с подключенными к нему компьютерами. Не нагруженным сегментом называется концентратор только с подключенными к нему другими концентраторами.

Это правило носит название «правило 5-4-3». Важным показателем работоспособности сети является коэффициент загрузки сегмента сети S:  $S = P * m_i / f$ ,

где P – количество компьютеров в сегменте сети

$m_i$  – количество кадров в секунду, отправляемых в сеть i-м узлом;

f – максимально возможная пропускная способность сегмента, равная, как было указано выше 148800 кадр/с.

Имитационное моделирование сети Ethernet и исследование её работы с помощью анализаторов протоколов показали, что при коэффициенте загрузки  $S > 0,5$  начинается быстрый рост числа коллизий и, соответственно, увеличивается время ожидания доступа к сети.

Рекомендуемая величина коэффициента загрузки  $S$  для сети, использующих стандарт Ethernet, должна быть:  $\leq 0,3$ .

Экспериментальные данные показали, что каждый из компьютеров передаёт в сеть в среднем от 100 до 600 кадров в секунду. Обычно берут значение 300. Таким образом, коэффициент загрузки сегмента равен:

$$S = 36 * 300 / 148800 = 0,0725.$$

Полученные данные свидетельствуют о том, что коэффициент загрузки сегмента сети не превышает рекомендуемое значение 0.3, соответственно, время ожидания будет минимальным.

Удвоенная задержка распространения сигнала (Path Delay Value, PDV) между двумя самыми удаленными друг от друга станциями сети не должна превышать 575 битовых интервалов.

Соблюдение этого требования обеспечивает своевременное обнаружение коллизий.

Рассчитаем значение PDV для нашего проекта:

- 1, Левый сегмент 1:  $15,3$  (база) +  $40 \text{ м} * 0,113/\text{м} = 19,8$
- 2, Промежуточный сегмент 2:  $42 + 30 \text{ м} * 0,113/\text{м} = 45,4$
- 3, Промежуточный сегмент 3:  $42 + 2 \text{ м} * 0,113 = 42,2$
- 4, Промежуточный сегмент 4:  $42 + 5 \text{ м} * 0,113 = 42,6$
- 5, Промежуточный сегмент 5:  $42 + 3 \text{ м} * 0,113 = 42,3$
- 6, Промежуточный сегмент 6:  $42 + 2 \text{ м} * 0,113 = 42,2$
- 7, Промежуточный сегмент 7:  $42 + 2 \text{ м} * 0,113 = 42,2$
- 8, Правый сегмент 8:  $165 + 2 \text{ м} * 0,113 = 165,2$

Таким образом, PDV сети равно:

$$PDV = 19,8 + 45,4 + 42,2 + 42,6 + 42,3 + 42,2 + 42,2 = 441,9.$$

Так как значение PDV меньше максимально допустимой величины 575, то эта сеть соответствует требованиям по величине максимально возможной задержки оборота сигнала.

Предприятие имеет прибыль в виде экономии, а ЗП и экономии налогов с фонда ЗП, которая составляет:

$$Pr = ЗП2 + Эн2 = 3041720000 + 1277522400 = 4319242400 \text{ р.}$$

Чистая прибыль предприятия рассчитывается по формуле:

$$Пч = Pr - Нпр,$$

где Нпр – налог на прибыль (33 % от суммы прибыли).

Подставив значения, получим:

$$\begin{aligned} Пч &= Пр - Нпр = Пр - Пр * 0,33 = \\ &= 4319242400 - 4319242400 * 0,33 = 2893892408 \text{ р.} \end{aligned}$$

Основные технико-экономические показатели спроектированной сети приведены в таблице 1.

Таблица 1 – Основные технико-экономические показатели проекта

Основные характеристики	Ед. изм.	Проект
Скорость передачи данных	Мбит/с	100 Мбит/с
Количество рабочих станций		36
Топология		смешанная
Среда передачи данных		витая пара
Пороговая граница коэффициента загрузки сети	%	0,0725
Защищенность от перегрузок	кВ	1,0 кВ электросеть 0,5 кВ сигнальная сеть
Основные характеристики	Ед. изм.	Проект
Стоимость внедрения проекта	рубель	87210200р.
Экономия заработной платы	рубель	2893892408р.
Срок окупаемости	год	0,03 года

### Литература

1. Максимов, Н.В. Компьютерные сети: учеб. пособие для вузов / Н.В. Максимов, И.И. Попов. – М.: ФОРУМ: ИНФРА-М, 2013. – 464 с.
2. Олифер, В.Г. Компьютерные сети. Принципы, технологии, протоколы: учеб. для вузов / В.Г. Олифер, Н.А. Олифер. – СПб.: Питер, 2014. – 944 с.
3. Ватаманюк, А.И. Создание, обслуживание и администрирование сетей / А.И. Ватаманюк. – СПб.: Питер, 2010. – 288 с.

**А.В. Купреева (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **Е.Е. Пугачёва**, ассистент

### **РАЗРАБОТКА ПРИЛОЖЕНИЯ ПРИ ПОМОЩИ СРЕДСТВ VISUAL FOXPRO**

Visual FoxPro (VFP) – объектно-ориентированный и процедурный язык программирования систем управления реляционными базами

данных, разработанный корпорацией Microsoft. Основой для данного программного продукта послужил язык программирования FoxPro. Относится к семейству языков xBase, разработанных на базе синтаксиса языка программирования dBase.

Visual FoxPro – это инструмент, который представляет собой визуальную среду создания систем управления реляционными базами данных. На сегодня финальной его версией является 9.0.

Система располагает сосредоточенным на работе с данными, объектно-ориентированным языком, который предлагает пользователям гибкий инструментарий. Он направлен на создание приложений на основе баз данных для индивидуального пользования, для функционирования в клиент-серверных средах и в глобальной сети.

В свое распоряжение пользователи получают все необходимое, чтобы уверенно управлять информацией – средства для упорядочивания таблиц, создания интегрированных систем управления базами данных, обработки запросов, а также инструменты полноценной разработки приложений для конечных потребителей.

Программа работает с большим множеством форматов.

Visual FoxPro – полнофункциональный язык визуального программирования, который объединяет в себе такие качества, как мощностъ средств, интерактивный режим работы, возможность создания приложений для управления реляционными базами данных, и, что не менее важно, этот язык прост в изучении и совместим с предыдущими версиями языка.

Работа начинается с создания проекта. Проект содержит в себе практически всю информацию о структуре базы данных: таблицы, форму, отчёты.

Первым шагом при разработке программного модуля, является создание проекта. *Project Manager*, позволяет объединять все элементы системы в единое целое, а так же обеспечивает доступ к программам, данным, объектам, экранам и т. д.

Для того, что бы создать проект в *Visual FoxPro*, нужно выбрать *File – New – Project* и выбираем *New file*. Появляется окно *Create* предлагающее нам сохранить проект, зададим ему имя, выберем куда сохранить и нажмём кнопку *Сохранить*. После перечисленных действий в рабочей области программы появится окно *Project Manager*.

Следующий этап – это создание таблиц. Для этого выбираем *File – New – Table* и выбираем *New file*. Появляется окно *Create* предлагающее нам сохранить таблицу, зададим таблице соответствующее имя, выберем папку для сохранения и нажмём кнопку *Сохранить*.

После нажатия кнопки *Сохранить*, появляется окно, имеющее вкладки: *Fields*, *Indexes* и *Table*. Выбираем первую вкладку *Fields*

и заполняем поля. В разделе *Name* вводим имена полей, в разделе *Type* задаём соответствующий тип и в разделе *Width* задаём длину поля. После заполнения таблицы нажимаем *OK*. После этого, заполняем таблицы нужными данными.

После того, как таблицы были созданы и заполнены, нужно их проиндексировать. Для этого можно воспользоваться окном *Project Manager*, где выбирается соответствующая таблица двойным нажатием на мыши, после чего появляется окно *Table Designer*, в котором будем выбирать вкладку *Indexes*. В данном окне в каждой таблице выбирается ключевое поле, по которому она будет в дальнейшем связываться, например, с дочерней таблицей. Таким образом, поступаем со всеми имеющимися в данном проекте таблицами.

После индексирования, связываем таблицы в базу данных. Что бы связать таблицы, нужно подвести к индексному ключу в таблице-родителе и удерживая нажатой правую кнопку мыши, переместить её на связанный ключ в дочерней таблице.

Для создания формы, выбираем в окне *Project Manager* вкладку *Documents*, выбираем пункт *Forms*, нажимаем кнопку *New – Form Wizard* (мастер форм). Затем в появившемся окне выбираем *One-to-Many Form Wizard* (мастер форм «один ко многим») и нажимаем *OK*. После проделанных действий нужно будет пройти шесть пунктов, в которых: выбираются поля, которые будут выводиться на экран, будет автоматически установлена связь по ключам в таблице, выбирается порядок сортировки и вид формы. В завершение выбираем *Save form and modify it in the Form Designer* (Сохранить форму и модифицировать её в конструкторе форм), вводим название, выбираем куда сохранить, нажимаем *Finish*.

Формы – это объекты контейнерного типа, которые имеют свои собственные уникальные свойства, события и методы. Формы содержат средства управления, которые нужны для разработки функциональных возможностей (например: ввод, просмотр, изменение таблиц и т. п.) различных приложений.

Завершающим этапом, является создание отчётов. Для создания отчёта, нужно открыть окно *Project Manager*, где можно выбрать вкладку *All* и в разделе *Documents – Report* нажимаем на кнопку *New*, после нажатия на кнопку *New*, в появившемся окне выбираем *Report Wizard*.

Затем в появившемся окне выбираем *One-to-Many Form Wizard* (мастер форм «один ко многим») и нажимаем *OK*. После проделанных действий нужно будет пройти шесть пунктов.

Отчет – предназначен для описания алгоритмов, при помощи которых пользователь сможет получать необходимые ему выходные

данные. Алгоритм формирования выходных данных описывается при помощи визуальных средств или с использованием встроенного языка. В реальной жизни объектам конфигурации Отчет соответствуют всевозможные таблицы выходных данных, сводных данных.

### Литература

1. Клепинин, В.Б. Visual FoxPro 9.0 / В.Б. Клепинин, Т.П. Агафонова. – СПб.: БХВ-Петербург, 2007. – 1216 с.
2. Гурвиц, Г.А. Разработка реального приложения с использованием Microsoft Visual FoxPro 9: учеб. пособие / Г.А. Гурвиц. – Хабаровск: ДВГУПС, 2007. – 198 с.
3. Бартенев, О. Microsoft Visual FoxPro. Учеб.-справ. пособие / О. Бартенев. – Россия: Диалог-МИФИ, 2005. – 672 с.
4. Мусина, Т. Visual FoxPro 9.0. Учеб. курс / Т. Мусина. – Россия: КОРОНА-Век, 2011. – 528 с.

**А.В. Купреева (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **Е.Е. Пугачёва**, ассистент

### РАЗРАБОТКА ПРОГРАММНОГО МОДУЛЯ «ОХРАНА ТРУДА» ПРИ ПОМОЩИ СРЕДСТВ VISUAL FOXPRO

Первый шаг – создание проекта. *Project Manager*, позволяет объединять все элементы системы в единое целое, а так же обеспечивает доступ к программам, данным, объектам, экранам и т. д.

Для того, что бы создать проект в Visual FoxPro, нужно выбрать *File – New – Project* и выбираем *New file*. Появляется окно *Create* предлагающее нам сохранить проект, зададим ему имя и нажмём кнопку *Сохранить* и появится окно *Project Manager*. Следующий этап – это создание таблиц.

После появляется окно, имеющее вкладки: *Fields, Indexes* и *Table*. Выбираем первую вкладку *Fields* и заполняем поля. В разделе *Name* вводим имена полей, в разделе *Type* задаём соответствующий тип и в разделе *Width* задаём длину поля. После, нажимаем *OK*. Затем заполняем таблицы данными.

На этом этапе были созданы таблицы: *spr\_period, spr\_soot\_f, spr\_konst, faktori, teh, spr\_rabotnik, spr\_vred*.

Далее создаются формы. Выбираем в окне *Project Manager* вкладку *Documents*, выбираем пункт *Forms*, нажимаем кнопку *New – Form Wizard*. Затем в появившемся окне выбираем *One-to-Many Form Wizard*



и нажимаем *OK*. После проделанных действий нужно будет пройти шесть пунктов, в которых: выбираются поля, которые будут выводиться на экран, будет автоматически установлена связь по ключам в таблице, выбирается порядок сортировки и вид формы. В завершение выбираем *Save form and modify it in the Form Designer*, вводим название, выбираем куда сохранить, нажимаем *Finish*.

На данном этапе созданы формы: справочник констант, справочник соотношения вредных производственных факторов должности и подразделению, справочник для добавления техпроцессов, оборудования.

Завершающим этапом, является создание отчётов. Для создания отчёта, нужно открыть окно *Project Manager*, где можно выбрать вкладку *All* и в разделе *Documents – Report* нажимаем на кнопку *New*, после нажатия на кнопку *New*, в появившемся окне выбираем *Report Wizard*.

На этом этапе созданы отчёты: перечень производственных факторов на рабочих местах в ОАО «Коминтерн», список профессий (должностей) подлежащих медосмотру, список работников ОАО «Коминтерн», подлежащих периодическому медицинскому осмотру на n-год и комплексная гигиеническая оценка условий труда работающих.

**А.И. Кучеров (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Д. Левчук**, канд. техн. наук, доцент

## **ПУТИ ОБЕСПЕЧЕНИЯ НАДЕЖНОСТИ ФУНКЦИОНИРОВАНИЯ УЗЛОВ ЛОКАЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ**

В процессе функционирования вычислительной системы возникают сбои и отказы как аппаратуры, так и программных средств входящих в состав этой вычислительной системы. Для своевременного реагирования на сбои и отказы возникающие в процессе функционирования вычислительной системы, необходимо доподлинно знать состав аппаратных и программных средств этой системы. При запуске вычислительной системы происходит ее самодиагностика или другими словами POST-диагностика (Power-On Self Test). POST – Программа, расположенная в микросхеме BIOS, загружается первой после включения компьютера. BIOS (Basic Input/Output System – базовая система ввода-вывода). Программа системного уровня, предназначенная для первоначального запуска вычислительной системы, настройки оборудования и обеспечения функций ввода/вывода. BIOS записывается

в микросхему постоянной памяти, которая расположена на системной плате.

Программа POST-диагностики определяет и проверяет установленное оборудование, настраивает устройства и готовит их к работе. При самотестировании, возможно, будет обнаружена неисправность оборудования, тогда процедура POST будет остановлена с выводом соответствующего сообщения или звукового сигнала. Если же все проверки прошли успешно, самотестирование завершается вызовом встроенной подпрограммы для загрузки операционной системы. Ну а если же программой будет выявлена серьезная ошибка, работа системы будет остановлена с выдачей звуковых сигналов, которые укажут на возникшую неисправность. У разных производителей системных плат от ведущих производителей BIOS, такие как: AMI BIOS, Award BIOS, Phoenix BIOS, существуют свои системы подачи сигналов, где набор коротких и длинных сигналов будет соответствовать определенной критической ошибке.

Но звуковое сопровождение не всегда присуще какой-либо критической ошибке и поэтому при процедуре самотестирования POST появление критической ошибки будет указываться сообщением диагностики.

POST-диагностика, является первым этапом обеспечения надежности функционирования узлов локальной вычислительной сети. Для более существенного повышения надежности эксплуатации вычислительной системы применяют и другие методы обеспечения надежности. Одним из наиболее эффективных методов обеспечения надежной работы вычислительной системы, является дублирование или резервирование ее узлов. В идеале все узлы вычислительной системы должны иметь двойника, который будет немедленно вступать в работу при выходе из строя основного узла. Если у каждого узла вычислительной системы будет не один, а два, три, четыре, и более дублирующих узлов. То надежность вычислительной системы возрастет в несколько раз, но ее цена возрастет многократно. А эти затраты не всегда оправданы. Для обеспечения безотказной работы вычислительной системы, очень часто достаточно проводить профилактические работы по ее обслуживанию.

Помимо надежности аппаратных средств надежность функционирования вычислительной техники напрямую связана с надежностью использования программных средств и привилегиями (правами) пользователей.

Пользователь может выполнять большое количество действий. Но не все из них пользователь имеет право и должен выполнять.

А информация может быть как общего, личного, так и служебного использования.

Для повышения надежности вычислительной системы администратор должен иметь возможность управлять правами пользователей локальной вычислительной сети и следить за выполнением их служебных обязанностей. Обеспечить эти возможности предназначено, как встроенное в операционную систему, так и другое системное программное обеспечение.

Каждый пользователь или группа пользователей в операционной системе обладают определенными правами. Действия, которые пользователь может выполнять в операционной системе, строго определены и описаны. В общем случае возможностей у пользователя много. Пользователь может выполнять большое количество различных операций, на которые он может иметь или не иметь прав. Эти операции связаны как с работой на локальном компьютере, так и при работе в локальной вычислительной сети.

Чем выше привилегии пользователя, тем выше у него права и соответственно возможности. Всеми правами в операционной системе обладают только администраторы системы. Для управления правами пользователей в операционной системе в настройках имеется возможность администрирования, где можно назначить права пользователя.

Пользователь может выполнять большое количество действий. Но не все из них пользователь имеет право и должен выполнять. А информация может быть как общего, личного, так и служебного использования.

Для повышения дисциплины руководство организаций и предприятий должно иметь возможность управлять правами пользователей локальной вычислительной сети и следить за выполнением их служебных обязанностей.

Современные операционные системы от версии к версии совершенствуются, отвечающие за безопасность. Войти в операционную систему возможно только зарегистрированному пользователю. Он должен знать зарегистрированное имя пользователя и его пароль. Если компьютер подключен к компьютерной сети с доменами в качестве рабочей станции, то операционная система потребует помимо имени и пароля, еще и имя домена. Только при совпадении этих трех составляющих пользователю будет разрешен вход в систему. То есть пользователь пройдет аутентификацию.

Обеспечить эти возможности предназначено как встроенное в операционную систему, так и другое системное программное обеспечение.

На рисунке 1 показана упрощенная схема защиты вычислительной техники от несанкционированного использования.



Рисунок 1 – Защита вычислительной техники от несанкционированного использования

Исходя из рисунка 1 видно, что надежность вычислительной системы так же зависит, от защиты вычислительной техники от несанкционированного использования, которая складывается из трех составляющих: административные средства, программные средства, аппаратные средства.

**П.Ю. Лаврук (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Кулинченко**, ст. преподаватель

## **ПРОБЛЕМА ПЕРЕПОЛНЕНИЯ ТАБЛИЦЫ MAC-АДРЕСОВ В ШИРОКОВЕЩАТЕЛЬНОМ ДОМЕНЕ СЕТИ УО «ГГУ ИМ. Ф. СКОРИНЫ»**

В настоящее время широко распространена технология BYOD (Bring Your Own Device), которая позволяет учащимся/работникам приносить свои устройства (ноутбуки, планшеты, смартфоны и т. д.) на место учебы / работы.

BYOD означает возможность использования в любом месте любого устройства, независимо от его владельца. Например, в прошлом для доступа к сети учебного заведения или Интернету учащиеся должны были использовать один из компьютеров учебного заведения. Эти устройства рассматривались, как правило, только как средства для работы в классе или в библиотеке. Расширенные возможности подключения с использованием мобильного и удалённого доступа к сети учебного заведения предоставляет учащимся огромную гибкость и более широкий спектр возможностей. Эта технология хорошо развита и в ГГУ «им. Ф. Скорины».

Но в связи с BYOD можно получить такую проблему как *Mac-table-overflow* (в переводе с англ. Переполнение таблицы MAC адресов). Проблема заключается в том, что каждое устройство при первоначальном подключении к сети ГГУ неизвестно коммутаторам и эти устройства рассылают ARP-запрос (широковещательный запрос, адресованный всем компьютерам в одном широковещательном домене). Суть запроса: «Компьютер с IP-адресом 10.0.0.1, сообщите компьютеру с IP-адресом 10.0.0.3 свой MAC-адрес». При этом на коммутаторе остается запись соответствия IP-адреса и MAC-адреса. Так постепенно заполняется таблица MAC-адресов, пока не достигнет физического предела.

При переполнении таблицы MAC-адресов коммутатор начинает работать как хаб, это означает, что, подключившись к любому порту коммутатора, можно перехватить весь трафик широковещательного домена, которому принадлежит порт. Это не касается тех записей в таблице коммутации, которые были до переполнения таблицы (т. к. пути к данным устройствам известны и трафик отправляется не широковещательно).

Широковещательный домен – область сети, в которой происходит обмен широковещательными сообщениями, и устройства могут отправлять сообщения непосредственно, без участия маршрутизатора.

В УО «ГГУ им. Ф. Скорины» широковещательный домен представлен всеми корпусами без исключения. Малая ограниченность домена является большой проблемой. Допустим было произведено подключение ноутбука в одном из кабинетов 4 корпуса и попытка «пропинговать» компьютер в том же корпусе - запрос пойдет по всему широковещательному домену.

Был проведен опыт в 4 корпусе университета, подключив ноутбук попробовали «пропинговать» компьютер, находящийся в кабинете 4-8. В результате этого в таблицу коммутации добавилось 40 соответствий, которые включали записи не только об устройствах корпуса № 4,

но и об устройствах из других корпусов. Не трудно предположить, что ограничение широковещательного домена приведет к удалению лишнего трафика. Одним способом является установка в каждом корпусе роутера, который бы ограничивал широковещательный домен до размера корпуса, но это того не стоит. Более простой и менее затратный способ – использовать VLAN.

VLAN (Virtual Local Area Network) – логическая («виртуальная») локальная компьютерная сеть, представляет собой группу хостов с набором требований, которые взаимодействуют так, как если бы они были подключены к широковещательному домену. Например, взяв 4 корпус и определив все устройства в 40 vlan, широковещательный трафик ограничится устройствами, которые находятся в vlan 40.

То есть, определив первый корпус как 10 vlan, 2 корпус как 20 vlan и так далее было бы проведено ограничение широковещательного домена и решена проблема переполнения таблицы MAC-адресов.

**Ю.Н. Лаврушик (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **В.Н. Леванцов**, ст. преподаватель

## **АВТОМАТИЗАЦИЯ РАБОЧЕГО МЕСТА ПРОДАВЦА-КОНСУЛЬТАНТА И МЕНЕДЖЕРА ТОРГОВОГО ЗАЛА**

В основе данного проекта лежит задача создания программного комплекса, охватывающего весь перечень прикладных задач, с которыми сталкиваются продавцы-консультанты и менеджеры торгового зала, а также построение системы, которая позволит эффективно использовать рабочее время, снизить количество используемой бумаги, а также ускорить и облегчить рабочий процесс.

Продавцы-консультанты занимаются консультированием покупателей и резервированием товара на торговом зале и центральном складе. Менеджеры торгового зала занимаются контролем резервов товара, аналитикой продаж, выкладки и оформлением заявок на доставку.

Проведя анализ существующих на сегодняшний день программных средств, было решено, что для автоматизации такого разностороннего проекта лучше всего подойдет программный продукт 1С: Розница 8 от компании 1С. Конфигурация «Розница» предназначена для автоматизации бизнес-процессов магазинов, которые могут входить в распределенную розничную сеть торгового предприятия. Может использоваться для автоматизации магазинов с большим количеством

рабочих мест, в том числе и в качестве кассовой программы. Первоначально 1С: Предприятие было предназначено для автоматизации бухгалтерского и управленческого учета (включая начисление заработной платы и управление кадрами), но сегодня этот продукт находит свое применение в областях, далеких от собственно бухгалтерских задач.

Конфигурация «Розница» рассчитана как на независимую, автономную работу, так и на взаимодействие с другими прикладными решениями. Двусторонние обмены данных с типовыми конфигурациями «Бухгалтерия предприятия», «Управление торговлей» позволяют создать программные комплексы, покрывающие потребности розничных предприятий различного масштаба.

В программе реализован многофирменный учет, при котором каждый склад магазина может быть отнесен к определенной организации. Для определенных торговых залов магазина может быть назначен учет ЕНВД.

«1С: Розница 8» автоматизирует типичные процессы розничного предприятия. Функциональные возможности программы в различных областях деятельности предприятия могут быть настроены в соответствии с принятой на предприятии технологии работы магазинов.

1С: Розница 8 позволяет решать следующие задачи:

1 Организовать ввод, хранение и своевременное распространение справочной информации по сети магазина.

2 Составить разностороннее описание целевой группы покупателей, магазинов и товаров для удобной работы сотрудников предприятия на всех технологических участках.

3 Управлять ассортиментом разноформатных розничных сетей и построить гибкое управление розничными ценами. Предоставляется возможность сбора и анализа информации о покупательских предпочтениях и настройка разнообразных стимулирующих скидочных программ.

4 Составлять заказы поставщику на основе анализа продаж и текущих товарных остатков с учетом исполнительности поставщика; осуществлять планирование и контроль оплаты поставок.

5 Поддерживать товарные остатки магазина в актуальном состоянии.

6 Технология штрихкодирования дает возможность быстро и точно учитывать движения товаров по складам магазина с необходимой для различных видов номенклатуры степенью детализации.

7 Управляющий магазином имеет возможность оптимизировать расписание работы сотрудников магазина с учетом сменного графика и нагрузки на кассовых узлах.

8 Регистрировать личные продажи.

9 Оформлять и хранить данные о доставках товара покупателям.

В программе поддерживается подключение оборудования, обеспечивающего технологию штрихкодирования на всех этапах работы с товаром: от создания карточки номенклатуры до продажи товара на кассовом узле.

База данных «Розница» хранит большой объем информации:

- весь перечень номенклатуры с характеристиками;
- информацию торговой точке;
- информацию о складах;
- данные о складских запасах;
- данные о сотрудниках торговой точки;
- данные о зарезервированном товаре;
- данные о проданном товаре;
- адресный классификатор;
- данные о доставках.

Пользовательский интерфейс обеспечивает отображения хранимых данных в удобном и наглядном для пользователя виде, что позволит с легкостью проводить над ними различные манипуляции.

Конфигурация базы данных содержит множество объектов. Одним из основных объектов является документ ЗаказПокупателя. Он содержит информацию о товаре, который в резерве, и о времени жизни резерва. Так же в документ записывается информация о торговой точке, складе резерва, продавце, который зарезервировал товар. Из этого документа распечатывается бланк резерва, предварительный чек, для покупателя.

Вторым документом по значимости является чек ККМ. В нем хранится информация о совершенной продаже с привязкой к заказу покупателя. При этом товар списывается с остатка, а резерв обнуляется.

Для менеджера торгового зала основным документом является заявка на доставку. Для оформления заказа на доставку необходимо знать информацию о товаре и правильно указывать данные по доставке. При этом нужно оформить несколько печатных форм – заявление и договор на доставку, этикетку на товар. С помощью созданного приложения этот процесс значительно облегчает работу менеджера. Адрес доставки он подбирает из специального классификатора улиц. Товар подбирается из списка продаж с отбором по чеку ККМ. При этом также улучшен процесс автоматизации самой доставки – сразу при оформлении менеджер видит загрузку транспортного средства с учетом всех оформленных на этот день доставок. По окончании оформления автоматически формируются все печатные формы, которые остается только распечатать.



Основным объектом конфигурации также является справочник Номенклатура, который хранит информацию о наименовании номенклатуры, её весе, объеме, краткой характеристике.

В конфигурацию включен весь перечень отчетов, с которыми могут столкнуться пользователи информационной системы. Например, отчет «Анализ доступности товаров» показывает зарезервированный и свободный к продаже остаток товара.

Т. к. вся конфигурация создана на основе «Управляемого приложения» в интерфейсе Такси, то при размещении конфигурации на веб-сервере, в ней можно будет находиться удаленно. Необязательно будет устанавливать специализированное программное обеспечение – достаточно web-браузера.

В ходе выполнения данного проекта была проделана работа по проектированию и разработке конфигурации для продавцов-консультантов и менеджеров торгового зала на основе конфигурации 1С: Розница версии 8.3. Большая часть реквизитов документов заполняется на основании справочников и перечислений, что позволяет сэкономить время на заполнение документов.

Для удобства пользователя был реализован удобный, наглядный и интуитивно понятный интерфейс. Всего было разработано две подсистемы, в которых отображается весь рабочий процесс продавца-консультанта и менеджера торгового зала.

Для запуска приложения пользователю понадобится пройти аутентификацию. Это позволит уменьшить количество рисков, связанных с искажением и кражей корпоративных данных злоумышленниками. По полномочиям сотрудников за каждым пользователем закреплены определенные роли на создание, чтение, редактирование и удаление данных из конфигурации.

## Литература

1. Баронов, В.В. Автоматизация управления предприятием / В.В. Баронов. – М.: ИНФРА, 2000. – 239 с.
2. Габец, А.П. Реализация прикладных задач в системе «1С:Предприятие 8.2» / А.П. Габец, Д.В. Козырев, Д.С. Кухлевский, Е.Ю. Хрусталёва. – М.: ООО «1С-Публишинг», 2012. – 714 с.
3. Гончаров, Д. И. Решение специальных прикладных задач в 1С:Предприятие 8.2 / Д.И. Гончаров, Е.Ю. Хрусталёва. – М.: ООО «1С-Публишинг», 2012. – 300 с.
4. Общее описание| 1С:Розница 8|1С:Предприятие 8 // 1С:Предприятие 8 [Электронный ресурс]. – 2014. – Режим доступа: <http://v8.1c.ru/retail/>

**Ю.Н. Лаврушик (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Леванцов**, ст. преподаватель

## **РЕШЕНИЯ НА БАЗЕ 1С:РОЗНИЦА 8**

Возможности автоматизации бизнес-процессов магазинов, которые могут входить в распределенную розничную сеть торгового предприятия, являются очень важными для организаций, занимающихся розничными продажами.

Конфигурация «Розница» предназначена для автоматизации бизнес-процессов магазинов, которые могут входить в распределенную розничную сеть торгового предприятия. Может использоваться для автоматизации магазинов с большим количеством рабочих мест, в том числе и в качестве кассовой программы.

Конфигурация «Розница» рассчитана как на независимую, автономную работу, так и на взаимодействие с другими прикладными решениями. Двусторонние обмены данных с типовыми конфигурациями «Бухгалтерия предприятия», «Управление торговлей» позволяют создать программные комплексы, покрывающие потребности розничных предприятий различного масштаба.

В программе реализован многофирменный учет, при котором каждый склад магазина может быть отнесен к определенной организации. Для определенных торговых залов магазина может быть назначен учет ЕНВД.

«1С:Розница 8» автоматизирует типичные процессы розничного предприятия. Функциональные возможности программы в различных областях деятельности предприятия могут быть настроены в соответствии с принятой на предприятии технологии работы магазинов.

Управление нормативно-справочной информацией. Функциональность программы обеспечивает ввод, хранение и своевременное распространение справочной информации по сети магазина. Широкие возможности программы позволяют составить разностороннее описание целевой группы покупателей, магазинов и товаров для удобной работы сотрудников предприятия на всех технологических участках.

Маркетинг. Программа позволяет маркетологу управлять ассортиментом разноформатных розничных сетей и построить гибкое управление розничными ценами. Предоставляется возможность сбора и анализа информации о покупательских предпочтениях и настройка разнообразных стимулирующих скидочных программ.

Запасы и закупки. Менеджеру предоставляется возможность составлять заказы поставщику на основе анализа продаж и текущих

товарных остатков с учетом исполнительности поставщика; осуществлять планирование и контроль оплаты поставок.

**Склад.** Функциональность раздела позволяет поддерживать товарные остатки магазина в актуальном состоянии. Технология штрихкодирования дает возможность быстро и точно учитывать движения товаров по складам магазина с необходимой для различных видов номенклатуры степенью детализации.

**Продажи.** Программа ориентирована на оформление розничных продаж потоковым методом. Гибко настраиваемый, интуитивно понятный графический интерфейс рабочего места кассира (РМК) позволяет быстро сформировать чек и оформить оплату покупки.

**Финансы.** Поддерживается учет различных средств оплаты розничных покупок: наличные, оплата платежной картой, подарочным сертификатом и с помощью заключения договора потребительского кредитования. Остатки наличных денежных средств и сертификатов учитываются по местам фактического хранения.

**Персонал.** Управляющий магазином имеет возможность оптимизировать расписание работы сотрудников магазина с учетом сменного графика и нагрузки на кассовых узлах. Поддерживается регистрация личных продаж.

В программе поддерживается подключение оборудования, обеспечивающего технологию штрихкодирования на всех этапах работы с товаром: от создания карточки номенклатуры до продажи товара на кассовом узле.

**Е.В. Леванцов (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **А.В. Воружев**, канд. техн. наук, доцент

## **ИСПОЛЬЗОВАНИЕ ИНТЕГРИРОВАННЫХ МЕДИАУСТРОЙСТВ**

Моноблок – тип исполнения техники, объединяющей несколько устройств в одном корпусе. Сама идея моноблока не нова. До компьютеров, было множество моноблочных устройств: радиолы, холодильные устройства. Моноблоки также используются в студийном освещении.

Наиболее близким устройством к моноблокам-компьютерам является Smart TV. В сущности Smart TV – это компьютер в форм-факторе телевизора, поддерживающий ряд возможностей обоих устройств (рисунок 1).



Рисунок 1 – Smart TV от компании Samsung

Моноблоки появились сравнительно недавно на рынке и поэтому их производители это крупные IT-компании, готовые пойти на риск вывода новой продукции на рынок. Это Apple, Lenovo, HP, ASUS, Sony, Samsung.

Моноблоки имеют целый ряд преимуществ и недостатков. Первое и неоспоримое преимущество моноблока заметно из его названия – это его собранность в едином корпусе (рисунок 2). То есть мы имеем в одном едином корпусе моноблока все элементы стандартного компьютера: экран, акустика, всевозможные порты ввода и вывода, дисковый привод, картридеры, камера, ТВ-тюнер, в некоторых моделях также присутствует WiFi и Bluetooth. И все это компактно расположено в едином корпусе-мониторе, размер которого обычно начинается от 18 дюймов, но есть и модели с 15 дюймовыми экранами. Это очень удобно, никаких проводов и подключений, кроме клавиатуры и мыши. А если вы будете использовать беспроводную мышь и клавиатуру, а также соединение с интернетом (как же сегодня без интернета) будет осуществлено через WiFi, то проводов станет еще меньше, а именно только один – это провод питания электроэнергией.

Из-за своего компактно размера моноблоки, выглядят стильно и привлекательно. Они являются «золотой жилой» для дизайнеров.

Во многие моноблоки устанавливаются сенсорные экраны, которые являются вполне приемлемой альтернативой.



Рисунок 2 – Моноблок iMac

Моноблоки – это великолепные мультимедийные центры. С большим и качественным экраном и хорошей встроенной акустикой, благодаря размерам моноблока, эти компьютеры очень удобны для просмотра фильмов и ТВ. Если сравнить с большим мультимедийным ноутбуком, то All-in-One PC выигрывает по всем параметрам – это и большая диагональ экрана и более качественный звук, так как размер моноблока больше ноутбука, то и встроить в него можно динамики большего размера, соответственно получаем более качественное звучание.

Из всего этого следует что моноблоки – одно из лучших решений для офиса, где компактность играет не последнюю роль. А производительность достаточна для работы с офисными приложениями.

Но за компактность и удобство приходится платить. All-in-One PC обладают целым рядом недостатков.

Первый связан с компактными размерами. Компьютеры моноблоки в сравнении с настольными ПК уступают в мощности и производительности, так как в ограниченном, небольшом корпусе невозможно уместить комплектующие (процессоры, видеокарты), которые используются

в настольных ПК и поэтому в этих системах используют мобильные версии комплектующих.

За ту же цену, что вы отдадите за all-in-one PC, вы можете купить классический настольный компьютер большей производительности и возможностей. За компактность и универсальность All-in-One PC придется заплатить больше, чем за аналогичный по характеристикам ПК.

Ещё один недостаток моноблоки «делят» с ноутбуками. Из-за компактных размеров и плотного расположения внутренних элементов возможности апгрейда системы сильно ограничены. Заменить и обновить возможно только быстросъемные элементы – оперативная память и жесткий диск.

Моноблоки хорошо проявляют себя как офисные компьютеры и мультимедийные центры, но как компьютеры, на которых ведётся работа с ресурсоёмкими приложениями, моноблоки уступают стационарным в равной ценовой категории.

**С.В. Леванцов (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В. Д. Левчук**, канд. техн. наук, доцент

## **ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ СЕРВИСА DATALEX TRAVEL DISTRIBUTION PLATFORM ПО ОНЛАЙН ЗАКАЗУ АВИАБИЛЕТОВ**

Программное обеспечение и услуги компании Datalex включают в себя цифровой рынок для более одного миллиарда покупателей, охватывающее каждый уголок земного шара используемое одними из самых инновационных в мире авиакомпаниями. Datalex является ведущим поставщиком электронной коммерции и розничных программных решений для многих крупнейших и наиболее прибыльных в мире авиакомпаний розничной торговли, и которая присутствует на каждом рынке и на каждом континенте.

При разработке системы были реализованы следующие функции, необходимые для эффективного использования как менеджментом авиакомпаний, так и простыми людьми, которые приобретают авиабилеты:

- интуитивно понятный интерфейс, благодаря которому заказчик может управлять даже самыми сложными поисками;
- расчёт стоимости и доступности для каждой возможной комбинации перелётов и сопутствующих для них продуктов;

- динамическая настройка цен на разных рынках, каналах поставок услуг и между членами альянса;
- эффективное управление миллионами запросов в секунду, не обременяющее компьютерные системы бронирования (Computer Reservation System);
- автоматизированный импорт тарифов и правил, полная поддержка системы консолидации и распределения тарифов на авиаперевозки, разработанной компанией АТРСО;
- приобретенные билеты могут быть возвращены, повторно перебронированы и пересчитаны в стоимости;
- возможность добавить в резервацию вспомогательные элементы и дополнительные услуги, такие как страховые услуги, бронирование автомобиля и отеля;
- система позволяет снизить количество дорогостоящих звонков в колл-центры.

Система представляет собой сайт, реализованный при помощи серверных и клиентских технологий и доступный из сети Интернет. В данной системе присутствуют все вышеперечисленные возможности.

**С.В. Леванцов (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Д. Левчук**, канд. техн. наук, доцент

## **АВТОМАТИЗАЦИЯ ЗАКАЗОВ АВИАБИЛЕТОВ С ПОМОЩЬЮ СИСТЕМЫ TRAVEL DISTRIBUTION PLATFORM**

Задача системы – создание и внедрение продукта, предназначенного для автоматизации заказов авиабилетов. Продукт внедряется на предприятие «AirTransat». Практический результат проекта – веб-приложение, доступное из сети компании и включающее клиентскую и серверную части. Автоматизированы процессы заказов на приобретение авиабилетов и сопутствующих услуг.

При разработке системы были реализованы следующие функции, необходимые для эффективного использования как менеджментом авиакомпаний, так и простыми людьми, которые приобретают авиабилеты:

- интуитивно понятный интерфейс;
- расчёт стоимости и доступности для каждой возможной комбинации перелётов и сопутствующих для них продуктов;
- динамическая настройка цен на разных рынках, каналах поставок услуг и между членами альянса;

- эффективное управление миллионами запросов в секунду;
- приобретенные билеты могут быть возвращены, повторно перебронированы и пересчитаны в стоимости;
- возможность добавить в резервацию вспомогательные элементы и дополнительные услуги, такие как страховые услуги, бронирование автомобиля и отеля.

Обмен данными между серверной и клиентской частями реализован при помощи REST-интерфейса. При этом данные передаются в формате XML по протоколу HTTP. На стороне сервера для создания REST-объектов для передачи используется технология Enterprise Java Beans. На клиентской стороне данные обрабатываются при помощи технологии Java Server Pages и фреймворком Struts 2.

Хотя рядовому пользователю использование приложения не кажется сложным, но внутренне оно представляет из себя огромную систему, разбитую на сотни модулей. Если не вдаваться в мелкие детали, то процесс достаточно прост:

- 1 Пользователь выбирает дату и маршрут полёта.
- 2 Система возвращает ему наиболее подходящий результаты по выбранным параметрам, и пользователь выбирает наиболее подходящий для него.
- 3 Пользователь вводит свои личные данные и информацию о системе оплаты заказа.
- 4 Система успешно завершает бронирование, если платёж прошёл.
- 5 Пользователь по желанию может посмотреть, изменить или отменить свою бронь.

На рисунке 1 показана диаграмма деятельности для системы бронирования авиабилетов.

Серверная часть приложения реализована на языке Java с использованием технологий Enterprise Java Beans и X-Doclets. В приложении использованы такие техники, как Java Beans (классы, написанные по определённым правилам и используемые для объединения нескольких объектов в один для удобной передачи данных), контроллеры (классы, используемые для обеспечения работы веб-сервиса по принципу MVC), DAO (Data Access Object – объекты, предоставляющие абстрактный интерфейс к типам базы данных и механизмам хранения), DTO (Data Transfer Object – шаблон проектирования, используемый для передачи данных между подсистемами приложения и не содержащий какого-либо поведения).

Клиентская часть приложения использует технологию JSP, фреймворк Struts 2, а также язык JavaScript с использованием библиотеки jQuery.



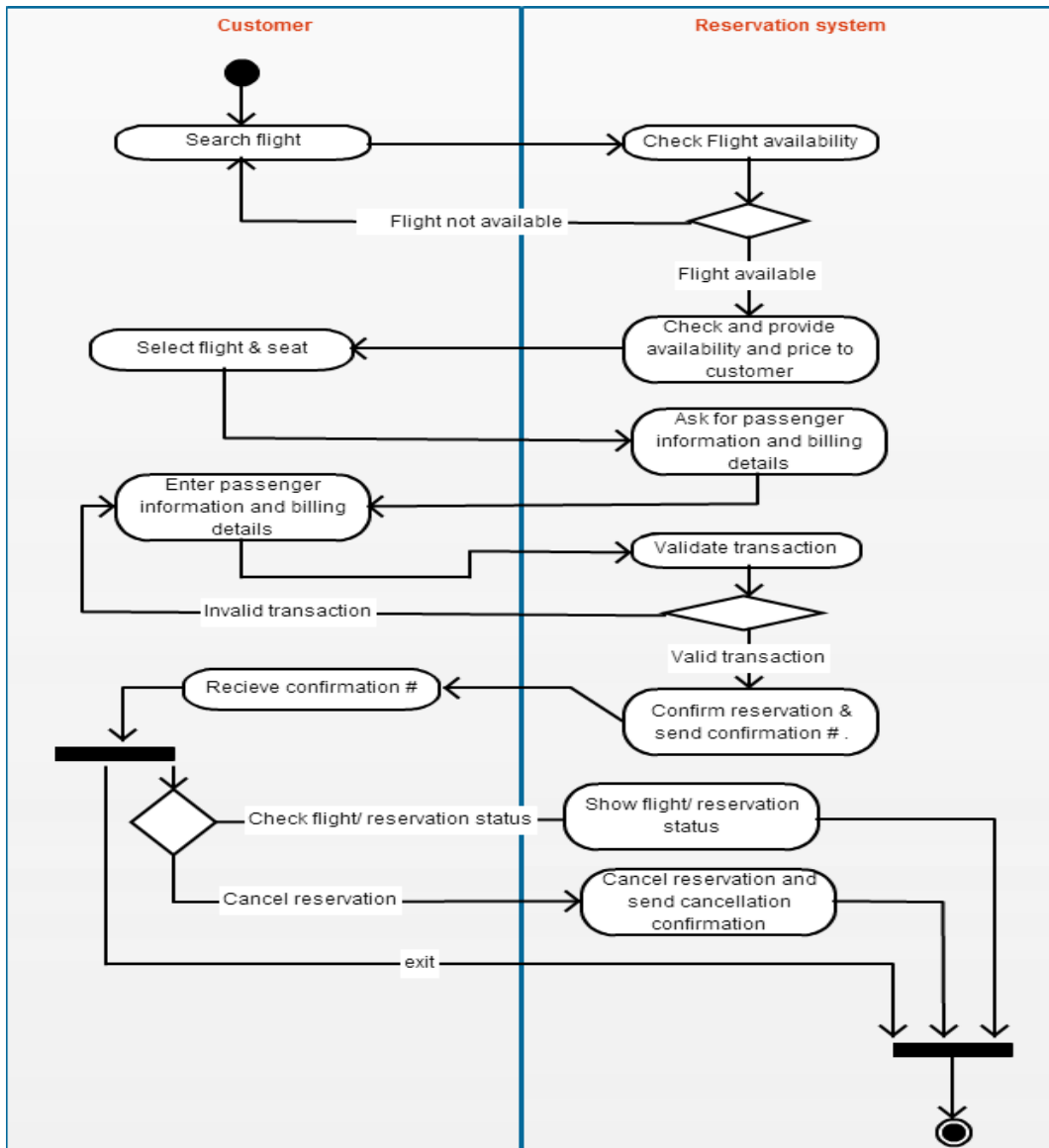


Рисунок 1 – Взаимодействие пользователя с системой

Клиентская и серверная части приложения обмениваются данными при помощи REST-интерфейса. В обобщённом виде схема обмена данными между клиентской и серверной частями выглядит следующим образом:

1 Клиент выполняет REST-запрос на сервер средствами фреймворков Struts и Servlet API.

2 Сервер при помощи EJB-сервисов обрабатывает запрос, выполняет требуемые действия с данными, после чего при помощи Data Access Object подготавливает ответ и отправляет его на клиентскую сторону.

3 Клиент принимает ответ от сервера и при помощи jsp тегов, JavaScript и HTML формируется страница, которая будет показана пользователю.

Для установки сервиса необходим мощный сервер, поддерживающий выполнение Java-приложений. Клиентская часть работает во всех современных браузерах: Opera, Mozilla Firefox, Google Chrome, Apple Safari, Microsoft Internet Explorer.

**А.М. Леоненко (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Леванцов**, ст. преподаватель

## **АНАЛИЗ СРЕДСТВ АВТОМАТИЗАЦИИ ЛИЦЕВОГО СЧЕТА В СРЕДЕ 1С: ПРЕДПРИЯТИЕ 8**

Автоматизация лицевого счета является необходимым компонентом любой зарплатно-кадровой конфигурации 1С: Предприятие 8, используемой на предприятии. Для автоматизации лицевого счета в среде 1С: Предприятие 8 используются унифицированная форма Т-54а. Это дополнительная форма, основной же формой является Т-54.

Они применяются для ежемесячного отражения сведений о заработной плате, выплаченной работнику в течение календарного года. Заполняются работником бухгалтерии. Форма Т-54 применяется для записи всех видов начислений и удержаний из заработной платы работника на основании первичных документов по учету выработки и выполненных работ, отработанного времени и документов на разные виды оплаты. Форма Т-54а применяется при автоматизированной обработке учетных данных средствами вычислительной техники (свт) с использованием специальных программ содержит условно-постоянные реквизиты, необходимые для расчета заработной платы. Экземпляр распечатки расчетного листка, содержащий данные о составных частях заработной платы, размерах и основаниях произведенных удержаний, об общей денежной сумме, подлежащей выплате, вкладывается (вклеивается) ежемесячно в лицевой счет работника на бумажном носителе. На оборотной стороне формы или вкладном листе дается расшифровка кодов (по системе кодирования, принятой в организации) различных видов выплат и удержаний. Также в эту форму вносятся общие данные о сотруднике. Такие как:

- дата приёма на работу;
- табельный номер;
- семейное положение;
- количество детей;
- дата увольнения;
- должность;
- название отдела.

Таким образом, лицевой счет несет в себе наиболее полную информацию о сотруднике, а именно хронологию его занятости на предприятии. Следовательно, автоматизация этого компонента бухгалтерской деятельности существенно упрощает и ускоряет работу специалистов.

В формировании лицевого счета задействовано множество компонентов и объектов конфигурации. В частности, такие которые отвечают за учёт разнообразных вычетов и удержаний применённых к сотрудникам предприятия. На современных предприятиях чаще всего используется лицевой счет, построенный именно на основе унифицированной формы Т-54а, так как она предполагает использование вычислительной техники, которая является неотъемлемым компонентом любого современного предприятия.

### Литература

1. Радченко, М.Г. 1С:Предприятие 8.2 Практическое пособие разработчика. Примеры и типовые приемы / М.Г.Радченко, Е.Ю. Хрусталева. – М.: ООО «1С-Паблишинг», 2009. – 874 с. : ил.
2. Кондраков, Н.П. Бухгалтерский учет : рек. М-вом образования РФ / Н.П. Кондраков. – М. : ИНФРА-М, 2008. – 716 с.
3. Селищев, Н.В. 1С:Бухгалтерия предприятия 8.2. Практическое пособие / Н.В. Селищев. – Кнорус, 2014. – 392 с.
4. Хрусталева, Е.Ю. Разработка сложных отчетов в 1С:Предприятие 8. Система компоновки данных / Е.Ю. Хрусталева. – 2-е изд. – М.: ООО «1С-Паблишинг», 2012. – 484 с.
5. Филатова, В.О. 1С для начинающих. Понятный самоучитель / В.О. Филатова. – СПб.: Питер, 2013. – 256 с.
6. Габец, А.П. Реализация прикладных задач в системе «1С:Предприятие 8.2» / А.П. Габец, Д.В. Козырев, Д.С. Кухлевский, Е.Ю. Хрусталёва. – М.: ООО «1С-Паблишинг», 2012. – 714 с.
7. Свободная энциклопедия Википедия [Электронный ресурс]. – 2014. – Режим доступа: <http://ru.wikipedia.org>. – Дата доступа: 05.12.2014.

**А.М. Леоненко (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Леванцов**, ст. преподаватель

### **АВТОМАТИЗАЦИЯ ЛИЦЕВОГО СЧЕТА В СРЕДЕ 1С: ПРЕДПРИЯТИЕ 8**

Одним из весомых условий успешной профессиональной деятельности на современном рынке труда среди экономистов, работников

финансовых служб, бухгалтеров и менеджеров является знание и свободное владение навыками работы с системой «1С: Предприятие 8.2». Корпоративная информационная система «1С: Предприятие 8.2» является гибкой настраиваемой системой, с помощью которой можно решать широкий круг задач в сфере автоматизации деятельности предприятий.

Автоматизация лицевого счета является необходимым компонентом любой зарплатно-кадровой конфигурации 1С: Предприятие 8, используемой на предприятии.

«1С: Предприятие» – программный продукт компании 1С, предназначенный для автоматизации деятельности на предприятии.

Первоначально, продукт «1С: Предприятие» был предназначен для автоматизации бухгалтерского учёта и управленческого учёта (включая начисление зарплаты и управление кадрами). Но сегодня этот продукт находит своё применение в областях, далёких от собственно бухгалтерских задач.

«1С: Предприятие» является универсальной системой автоматизации экономической и организационной деятельности предприятия. Поскольку такая деятельность может быть довольно разнообразной, система «1С: Предприятие» может «приспосабливаться» к особенностям конкретной области деятельности, в которой она применяется. Для обозначения такой способности используется термин конфигурируемость, то есть возможность настройки системы на особенности конкретного предприятия и класса решаемых задач.

Это достигается благодаря тому, что «1С: Предприятие» – это не просто программа, существующая в виде набора неизменяемых файлов, а совокупность различных программных инструментов, с которыми работают разработчики и пользователи. Логически всю систему можно разделить на две большие части, которые тесно взаимодействуют друг с другом: конфигурацию и платформу, которая управляет работой конфигурации. Платформа обеспечивает работу конфигурации и позволяет вносить в неё изменения или создавать собственную конфигурацию.

Существует одна платформа и множество конфигураций. Для функционирования какого-либо прикладного решения всегда необходима платформа и какая-либо (одна) конфигурация. Сама по себе платформа не может выполнить никаких задач автоматизации, так как она создана для обеспечения работы какой-либо конфигурации. То же самое с конфигурацией: чтобы выполнить те задачи, для которых она создана, необходимо наличие платформы, управляющей её работой.

Для того чтобы обеспечить работу, система «1С: Предприятие» имеет различные режимы работы: 1С: Предприятие и Конфигуратор.

Режим 1С: Предприятие является основным и служит для работы пользователей системы. В этом режиме пользователи вносят данные, обрабатывают их и получают итоговые результаты.

Режим Конфигуратор позволяет выполнять изменение существующих прикладных решений и создание новых, а также выполнять действия по администрированию информационной базы. Именно этот режим и предоставляет инструменты, необходимые для модификации существующей или создания новой конфигурации.

Конфигуратор предоставляет разработчику все необходимые инструменты для «визуального» описания структуры прикладного решения, создания форм диалогов и выходных документов, механизмов работы пользователей с данными (интерфейсов) и прав доступа различных групп пользователей к различной информации. Кроме этого конфигуратор позволяет создать и настроить взаимодействие различных элементов прикладного решения друг с другом и написать модули на встроенном языке для обработки различных событий и реализации специфических алгоритмов взаимодействия, обработки входных и выходных данных.

Для автоматизации лицевого счета в среде 1С: Предприятие 8 используются унифицированная форма Т-54а. Это дополнительная форма, основной же формой является Т-54.

Они применяются для ежемесячного отражения сведений о заработной плате, выплаченной работнику в течение календарного года. Заполняются работником бухгалтерии. Форма Т-54 применяется для записи всех видов начислений и удержаний из заработной платы работника на основании первичных документов по учету выработки и выполненных работ, отработанного времени и документов на разные виды оплаты. Форма Т-54а применяется при автоматизированной обработке учетных данных средствами вычислительной техники (свт) с использованием специальных программ содержит условно-постоянные реквизиты, необходимые для расчета заработной платы. Экземпляр распечатки расчетного листка, содержащий данные о составных частях заработной платы, размерах и основаниях произведенных удержаний, об общей денежной сумме, подлежащей выплате, вкладывается (вклеивается) ежемесячно в лицевой счет работника на бумажном носителе. На оборотной стороне формы или вкладном листе дается расшифровка кодов (по системе кодирования, принятой в организации) различных видов выплат и удержаний. Также в эту форму вносятся общие данные о сотруднике. Такие как:

- дата приёма на работу;
- табельный номер;
- семейное положение;
- количество детей;
- дата увольнения;
- должность;
- название отдела.

Таким образом, лицевой счет несет в себе наиболее полную информацию о сотруднике, а именно хронологию его занятости на предприятии. Следовательно, автоматизация этого компонента бухгалтерской деятельности существенно упрощает и ускоряет работу специалистов.

В формировании лицевого счета задействовано множество компонентов и объектов конфигурации. В частности, такие которые отвечают за учёт разнообразных вычетов и удержаний применённых к сотрудникам предприятия. На современных предприятиях чаще всего используется лицевой счет, построенный именно на основе унифицированной формы Т-54а, так как она предполагает использование вычислительной техники, которая является неотъемлемым компонентом любого современного предприятия.

### Литература

1. Радченко, М.Г. 1С:Предприятие 8.2 Практическое пособие разработчика. Примеры и типовые приемы / М.Г.Радченко, Е.Ю. Хрусталева. – М.: ООО «1С-Паблишинг», 2009. – 874 с. : ил.
2. Кондраков, Н.П. Бухгалтерский учет : рек. М-вом образования РФ / Н.П. Кондраков. – М. : ИНФРА-М, 2008. – 716 с.
3. Селищев, Н.В. 1С:Бухгалтерия предприятия 8.2. Практическое пособие / Н.В. Селищев. – Кнорус, 2014. – 392 с.
4. Хрусталева, Е.Ю. Разработка сложных отчетов в 1С:Предприятие 8. Система компоновки данных / Е.Ю. Хрусталева. – 2-е изд. – М.: ООО «1С-Паблишинг», 2012. – 484 с.
5. Филатова, В.О. 1С для начинающих. Понятный самоучитель / В.О. Филатова. – СПб.: Питер, 2013. – 256 с.
6. Габец, А.П. Реализация прикладных задач в системе «1С:Предприятие 8.2» / А.П. Габец, Д.В. Козырев, Д.С. Кухлевский, Е.Ю. Хрусталёва. – М.: ООО «1С-Паблишинг», 2012. – 714 с.
7. Свободная энциклопедия Википедия [Электронный ресурс]. – 2014. – Режим доступа: <http://ru.wikipedia.org>. – Дата доступа: 05.12.2014.

**А.Н. Лещинский (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **В.Н. Кулинченко**, ст. преподаватель

## **СИСТЕМА УЧЕТА И КОНТРОЛЯ ЗАЯВОК В ГОМЕЛЬСКОМ ФИЛИАЛЕ РУП «БЕЛПОЧТА»**

В настоящее время почти в любой организации у сотрудников и отделов будут возникать проблемы с компьютерами или же оргтехникой. В данном случае можно позвонить специалисту для решения этой проблемы, но его может не оказаться на месте. При этом в некоторых организациях, заявки регистрируются через приемную, что занимает продолжительного количества времени.

Автоматизированная система «Учет и контроль заявок» является необходимым инструментом в работе любого предприятия. Она позволяет быстро навести порядок в системе учета и обеспечить своевременное выполнение заявок (заказов). С ее помощью на предприятии осуществляется заказ технического оборудования, а также уведомления сотрудников служб о конкретных проблемах. На основе анализа данных системы учета за определенный период времени, начальники отделов (служб) могут планировать дальнейшую работу своего отдела и назначать задания свободным специалистам.

Эффективный учет заявок достигается за счет того, что система имеет свою базу данных, в которую заносятся справочные материалы, каталоги, адреса, ФИО пользователей и номера телефонов. Поэтому, данная система «Учета и контроля заявок» состоит из нескольких программных модулей, которые устанавливаются на компьютеры служб, имеющие возможность подачи заявки, на компьютеры, которые должны непосредственно выполнять заявку (заказ), а также компьютеры начальников служб. В последнем случае это необходимо для осуществления текущего контроля работы сотрудников и возможности немедленного получения информации.

Автоматизированный контроль за исполнением заявок позволяет быстро и удобно получить доступ к тем заявкам, которые выполнены с опозданием, либо же вообще не выполнены. Произвести ручной или же бумажный контроль за исполнением заявок на некоторых предприятиях довольно сложно. При автоматизированном контроле, можно получить доступ к любой заявке, любого отдела и за любой период времени, тем самым отследить ход выполнения этой заявки отделом (исполнителем).

Программная система «Учет и контроль заявок» позволяет упорядочить процесс, контролировать ход выполнения заявок (заказов),

обеспечить структурированное хранение данных, представлять необходимую отчетность и полностью автоматизировать деятельность диспетчерской, через которую проходит огромное количество заявок.

Программная система состоит из трех модулей: «Модуль заявителя», «Модуль контроля» и «Модуль исполнителя». Все модули являются настраиваемыми и имеют окно авторизации.

Этот программный продукт является универсальным, так как может быть установлен абсолютно в любой организации, соответственно он не имеет конкретной предметной области. Его можно применять там, где необходимо подавать заявки (заказы) для последующего их быстрого исполнения и контроля.

При использовании этого приложения повышается производительность и эффективность труда. Сокращается время обрабатываемой информации, так как заявки для исполнения попадают сразу к нужному исполнителю, что гарантирует быстроту их исполнения и эффективный контроль. Принцип работы системы показан на рисунке 1.

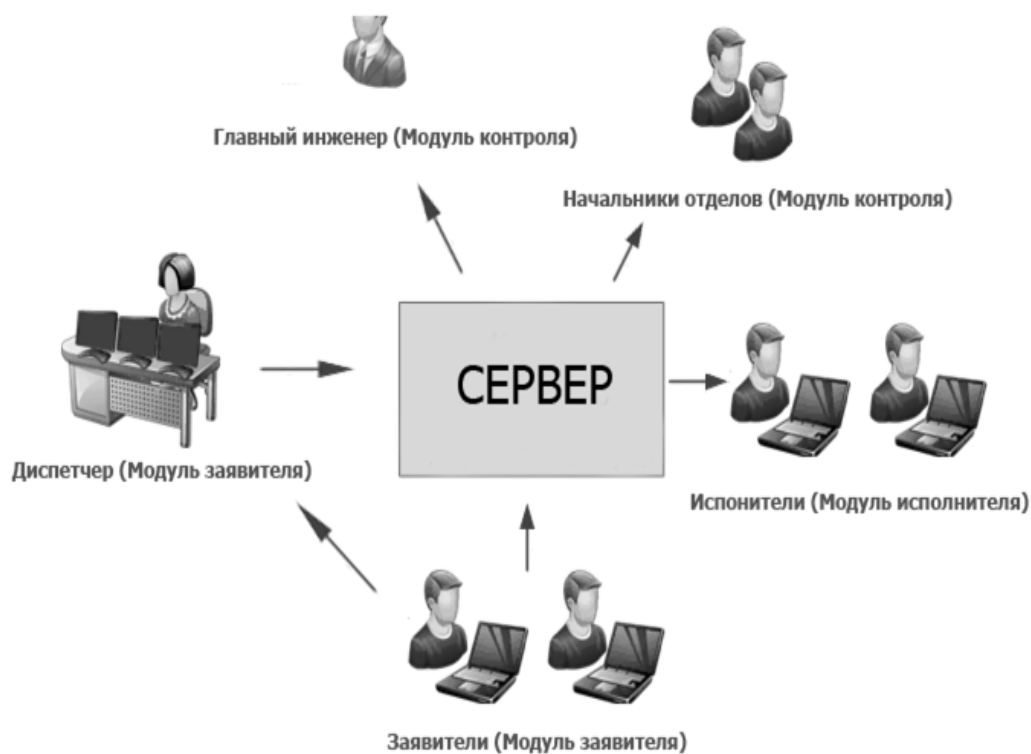


Рисунок 1 – Принцип работы системы «Учет и контроль заявок»

Программная подсистема имеет простой и понятный графический интерфейс, за счет этого в ней могут работать пользователи с базовыми знаниями персонального компьютера. Программа обрабатывает большие объемы данных и быстро предоставляет нужную информацию и отчеты.



## Литература

1. Душан, П. Microsoft SQL Server 2012. Руководство для начинающих / П. Душан. – СПб.: БХВ-Петербург, 2013 – 110 с.
2. Осипов, Д.Л. Базы данных и Delphi. Теория и практика / Д.Л. Осипов. – СПб.: БХВ-Петербург, 2011 – 220 с.
3. Осипов, Д.Л. DelphiXE2 / Д.Л. Осипов. – СПб.: БХВ-Петербург, 2012. – 303 с.

**А.Н. Лещинский (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **В.Н. Кулинченко**, ст. преподаватель

### **НЕОБХОДИМОСТЬ АВТОМАТИЗАЦИИ ПРОЦЕССА УЧЕТА И КОНТРОЛЯ ЗАЯВОК В ГОМЕЛЬСКОМ ФИЛИАЛЕ РУП «БЕЛПОЧТА»**

В настоящее время оперативная обработка потока входящих заявок по предоставлению информационно-технологических услуг в современных компаниях представляет собой непростую задачу. В основном проблемы связаны с отсутствием программного обеспечения, позволяющего автоматизировать существенные объемы ручной работы и упорядочить перечень запросов клиентов.

Анализируя структуры государственных учреждений, частных и государственных компаний можно заметить, что среднее количество компьютерных рабочих мест в них составляет около трехсот единиц. Помимо этого, существует необходимость обслуживания сетевого оборудования и телефонных сетей.

В связи с этим возникает необходимость создания системы, обеспечивающей учет оборудования, а также осуществляющей информационное сопровождение процесса обслуживания объектов. Внедрение такой системы существенно ускоряет процесс выполнения заявок, исключает возможности их утери, позволяет осуществлять контроль за исполнением работ, а также максимально объективно отслеживать состояние объектов материально-технической базы предприятия.

Основными задачами, решаемыми информационно-технологическими отделами современных организаций, являются:

- получение информации о состоянии и количестве оборудования, о распределении оборудования по подразделениям организации, об установленных в оборудовании комплектующих частях и программном обеспечении;
- учет перемещения оборудования внутри организации;

- учет оборудования на рабочих местах и регистрация изменений, произошедших после ремонта оборудования;
- регистрация поврежденного оборудования, формирование заявок на ремонт, а также фиксация этапов обработки заявок и стадий их выполнения;
- учет проведения ремонтных (регламентных) работ;
- учет имеющихся в организации программных продуктов.

Таким образом, автоматизируя данный процесс, можно повысить эффективность и производительность труда на любом предприятии, где будет внедрена данная программная подсистема.

**Ю.В. Лишко (УО БрГТУ, Брест)**

Науч. рук. **В.И. Хвещук**, канд. техн. наук, доцент

## **ВОПРОСЫ РЕАЛИЗАЦИИ МУЛЬТИПРОЦЕССИНГОВОЙ СИСТЕМЫ**

Широкое распространение Интернета и использование электронных платежных систем позволяет пользователям в реальном масштабе времени просматривать и заказывать товары и услуги. Платёжные системы позволяют проводить авторизацию и процессинг платежей при помощи банковских карточек.

Анализ рынка CRM-систем (Salesforce, 1С, MS Dynamics) показывает, что возможностей существующих систем уже недостаточно. Необходимо, чтобы их CRM-системы позволяли работать еще и в смежных сферах. Например, трейнинг (онлайн с учителем, посредством видео), анализ рабочей нагрузки подчиненных, организация баг-трекеров прямо в системе или же монетизация отношений.

Предлагаемая система позволяет пользователям проводить оплату прямо в CRM-системе Salesforce, где будет храниться необходимая информация. Для этой системы абсолютно не имеет значения, какого рода отношения будут монетезированы – это лишь платежное средство, которое позволит внедрить оплату услуг в уже устоявшийся рабочий процесс.

Проект разрабатывается на основе облачной платформе Force.com CRM-системы Salesforce.com. Для управления реляционной базой данных (БД) используется Database.com. Программирование реализуется на языке Apex. В качестве интерфейса выступают VisualForce страницы. Одним из основных преимуществ перед другими аналогами является поддержка стандарта PCI Compliance.

Создаваемая система построена на основе клиент-серверной архитектуры и состоит из следующих компонентов: подсистема платежей обеспечивает администрирование платежных операций; подсистема БД выполняет формирование и администрирование справочников пользователей и настроек; подсистема документирования обеспечивает формирование отчетов о проведенных транзакциях за сутки / неделю и другие; подсистема администрирования и управление обработкой внешних запросов.

Данная работа находится на стадии завершения реализации и выполняется в рамках дипломного проекта.

**А.А. Лосев (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **В.Н. Кулинченко**, ст. преподаватель

### **МОДЕРНИЗАЦИЯ ЛВС ОАО «ГОМЕЛЬСТЕКЛО»**

ОАО «Гомельстекло» является крупнейшим производителем стекольной продукции в Беларуси. Для успешной работы такому крупному предприятию не обойтись без современных технологий и, в особенности, без локальной вычислительной сети внутри самого предприятия.

Компьютерная сеть – это система объединенных между собой компьютеров, а также, возможно, других устройств, которые называются узлами (рабочими станциями) сети. Все компьютеры, входящие в сеть соединены друг с другом и могут обмениваться информацией.

Темп прогресса сейчас очень велик, выходит новое оборудование, создаются новые стандарты и т. п., увеличиваются потребности людей. И пользователи сети не являются исключением. Поэтому модернизация сети предприятий неизбежна.

Модернизация – это усовершенствование, улучшение, обновление объекта, приведение его в соответствие с новыми требованиями и нормами, техническими условиями, показателями качества.

В данной работе был проведён анализ локально вычислительной сети завода ОАО «Гомельстекло». Сеть предприятия организована по топологии «Звезда» с элементами топологии «Кольцо».

Звезда – базовая топология компьютерной сети, в которой все компьютеры сети присоединены к центральному узлу (обычно коммутатор), образуя физический сегмент сети. Подобный сегмент сети может функционировать как отдельно, так и в составе сложной сетевой топологии (как правило, «дерево»). Весь обмен информацией

идет исключительно через центральный компьютер, на который таким способом возлагается очень большая нагрузка, поэтому ничем другим, кроме сети, он заниматься не может. Сеть предприятия представлена на рисунке 1.

Сервер находится в здании вычислительного центра. В ходе анализа технического комплекса сервера нарушений выявлено не было, всё оборудование удовлетворяет современным стандартам и требованиям. Однако были выявлены недочёты в программном комплексе. Серверная операционная система Novell NetWare, используемая на предприятии, не поддерживается разработчиком с 2009 г. Ввиду этого был осуществлён переход на современную серверную ОС MS Windows Server 2012 R2.

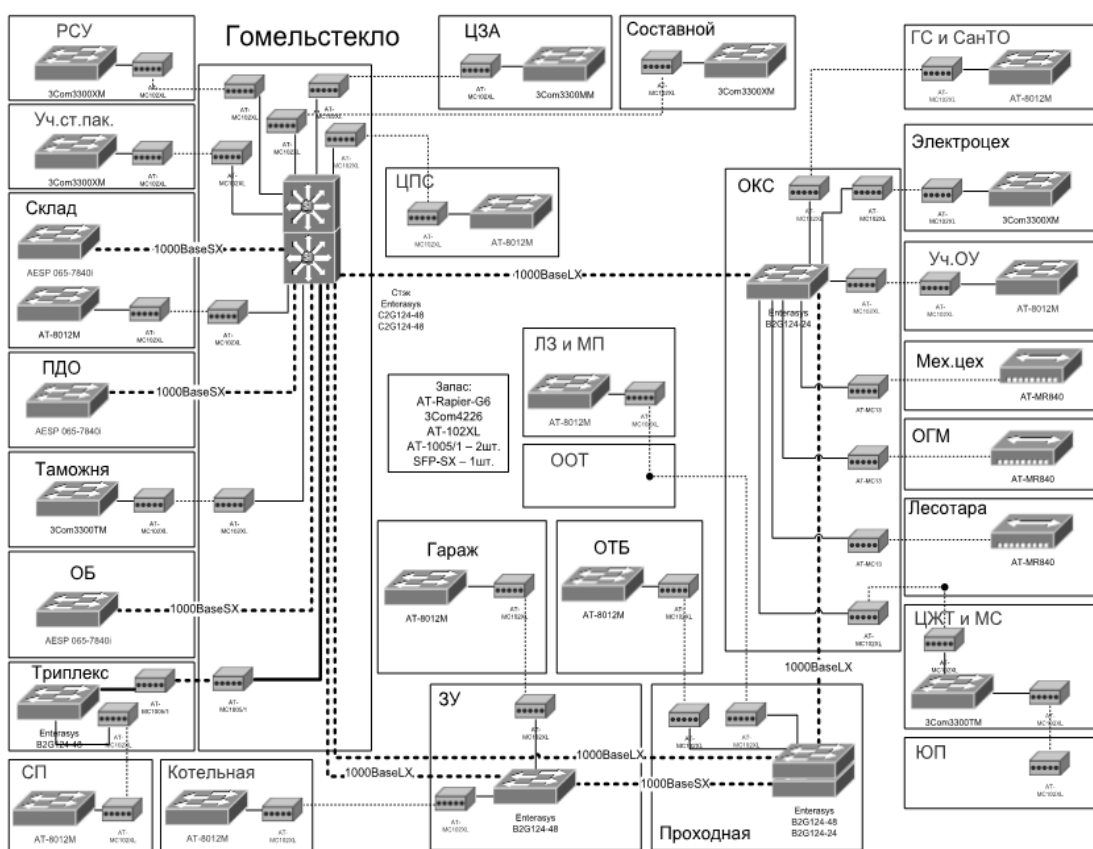


Рисунок 1 – Сеть ОАО «Гомельстекло»

Active Directory («Активный каталог», AD) – LDAP-совместимая реализация службы каталогов корпорации Microsoft для операционных систем семейства Windows NT. Active Directory позволяет администраторам использовать групповые политики для обеспечения единообразия настройки пользовательской рабочей среды, разворачивать программное обеспечение на множестве компьютеров через групповые политики или посредством System Center Configuration Manager,

устанавливать обновления операционной системы, прикладного и серверного программного обеспечения на всех компьютерах в сети, используя службу обновления Windows Server. Active Directory хранит данные и настройки среды в централизованной базе данных. Сети Active Directory могут быть различного размера: от нескольких десятков до нескольких миллионов объектов.

Были настроены службы Active Directory, т. е. осуществлена установка роли AD (доменные службы AD), настройка доменных служб AD, добавление новых пользователей и ввод компьютера в домен.

Далее анализ сети осуществлялся в здании заводоуправления, т. к. там сконцентрировано наибольшее количество оборудования. Здесь были выявлены следующие недостатки: использование морально устаревшего оборудования (сетевые концентраторы и слабые ПК), отдел охраны труда не имеет доступа в сеть. Для устранения этих недостатков сетевые концентраторы были заменены на сетевые коммутаторы. Выборочно заменены ПК на более современные и мощные. Проведена сеть в отдел охраны труда. Так же, в отделе охраны труда, из нескольких ПК и проектора был создан компьютерный класс для проведения аттестаций, семинаров и т. п.

Информационные розетки были установлены на стене на высоте 0,3–0,9 м от уровня пола по месту. Прокладка кабелей UTP в коридорах и комнатах с подвесными потолками выведена за подвесными потолками в трубах ПВХ50. Спуски кабелей UTP к розеткам выполнены в К25х16. Устройство ввода волоконно-оптического кабеля было закреплено на стене на высоте 1,5 м от уровня пола по месту.

Прокладка информационных кабелей была выполнена на расстоянии не менее 0,2 м от кабелей сети электропитания.

Маркировка розеток и соответствующих портов соединительных панелей распределительного устройства была выполнена в соответствии с приведенными нормами в рабочих чертежах и таблицах соединений.

Защитное заземление проложено кабелем марки ВВГЗ П 1х6.0 от проектируемого распределительного устройства до шины заземления (РЕ) существующего РП (ЩС).

После производства работ по прокладке кабельной сети все линии протестировались на соответствие техническим характеристикам, предъявляемым к линиям класса D IS011801, прибором, соответствующим требованиям TSB67 Level II.

Модернизация в других зданиях предприятия может быть осуществлена на примере здания заводоуправления, т. к. особого различия в программном и техническом комплексе нет.

**А. А. Лосев, (ГГУ им. Ф. Скорины, Гомель)**  
Науч. рук. **В.Н. Кулинченко, ст. преподаватель**

## **ПРОЕКТ МОДЕРНИЗАЦИИ ЛВС ЗДАНИЯ ЗАВОДОУПРАВЛЕНИЯ ОАО «ГОМЕЛЬСТЕКЛО»**

ОАО «Гомельстекло» является крупнейшим производителем стекольной продукции в Беларуси. Для успешной работы такому крупному предприятию не обойтись без современных технологий и, в особенности, без локальной вычислительной сети внутри самого предприятия.

В связи с этим было проведено предпроектное обследование локальной вычислительной сети завода ОАО «Гомельстекло». Сервер находился в здании вычислительного центра. В ходе анализа технического комплекса сервера нарушений выявлено не было, всё оборудование удовлетворяло современным стандартам и требованиям. Однако были выявлены недочёты в программном комплексе. Серверная операционная система Novell NetWare, используемая на предприятии, не поддерживалась разработчиком с 2009 г. Ввиду этого был осуществлён переход на современную серверную ОС MS Windows Server 2012 R2. Так же были настроены службы Active Directory (AD), т. е. осуществлена настройка доменных служб AD, добавление новых пользователей и ввод компьютера в домен.

Далее анализ сети осуществлялся в здании заводоуправления, т. к. там сконцентрировано наибольшее количество оборудования. В отделах здания заводоуправления были выявлены следующие недостатки: использование морально устаревшего оборудования (сетевые концентраторы и архаичные рабочие станции), отдел охраны труда не имел доступа в сеть.

Для устранения этих недостатков были проведены следующие работы:

- сетевые концентраторы были заменены на сетевые коммутаторы;
- проведена выборочная замена рабочих станций на более современные и мощные;
- проведена сеть в отдел охраны труда.

Так же, в отделе охраны труда, из нескольких ПК и проектора был создан компьютерный класс для проведения аттестаций и семинаров.

Модернизация в других зданиях предприятия может быть осуществлена на примере здания заводоуправления, т. к. особого различия в программном и техническом комплексе остальных зданий завода нет.

## ПРИМЕНЕНИЕ ГРАФЕНА

Графен – это первый доступный 2-D атомный «кристалл», сочетающий в себе уникальные свойства – высокую механическую прочность, электро- и теплопроводность, непроницаемость для газов, прозрачность. Этими особенностями не может похвастаться ни один из известных сплавов, что делает графен привлекательным материалом для многих электронных разработок. Его применяют в медицине, электронике, электроэнергетике и других областях.

В настоящее время в области применения графена ведутся разработки в следующих направлениях:

Высокочастотные транзисторы. Подвижность электронов в графене гораздо больше, чем в кремнии, поэтому цифровые элементы из графена обеспечивают более высокую частоту работы. Некоторые компании уже заявляли об успехах в этой области. Так, транзисторы IBM работают на частоте 26 ГГц и имеют размер около 240 нм (рисунок 1) [1].

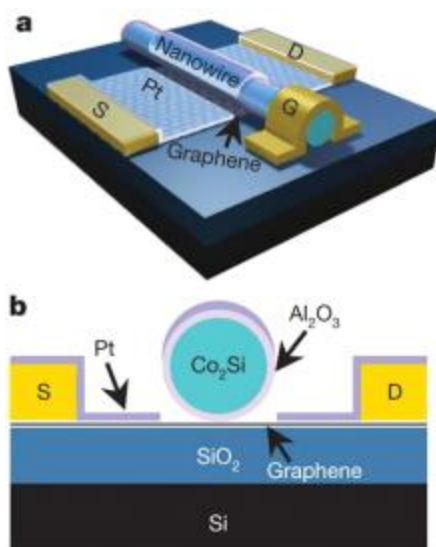


Рисунок 1 – Строение графенового транзистора

Микросхемы памяти. Прототип нового типа запоминающего устройства состоит всего из 10 атомов графена. Во время лабораторных тестов группе удалось создать кремниевые модули, на которых были размещены 10 атомарных слоев графена. В итоге графеновый слой получил толщину около 5 нм. В новых экспериментальных модулях базовые ячейки хранения информации примерно в 40 раз меньше

ячеек, используемых в самых современных 20-нм модулях NAND-памяти (рисунок 2) [2].

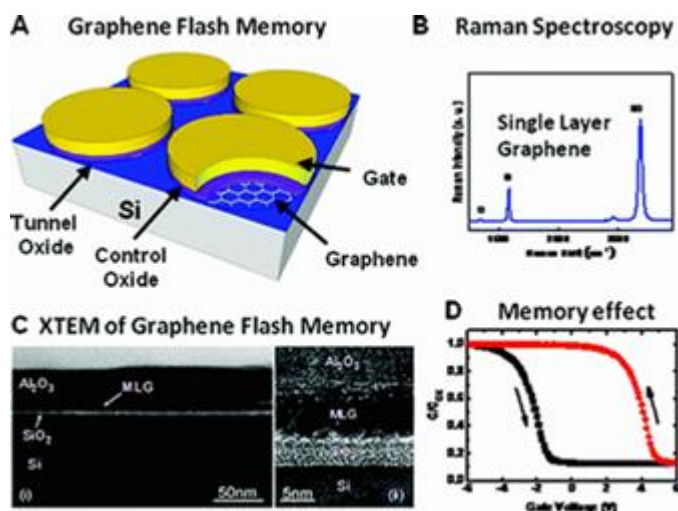


Рисунок 2 – Ячейка флэш-памяти на основе графена

Электроды для суперконденсаторов. Проводимость графеновых электродов превышает 1700 См/м, тогда как у электродов на активированном угле она составляет лишь 10–100 См/м. Благодаря высокой механической прочности LSG-электроды могут использоваться в суперконденсаторах без связующих элементов или токоприемников, что упрощает конструкцию и снижает себестоимость изготовления (рисунок 3) [3].

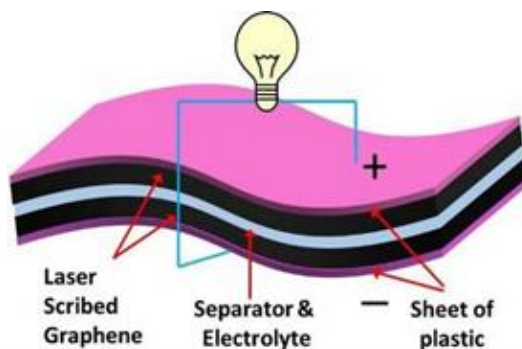


Рисунок 3 – Графеновый суперконденсатор (ионистор)

Кроме того, было установлено, что графен пропускает до 98 % света. Это значительно выше показателя пропускания лучших материалов из ИТО (82–85 %). Графен обладает высокой электропроводностью, что позволяет использовать его для создания прозрачных электродов, управляющих поляризацией и состоянием жидких кристаллов [4].

Другая группа исследователей недавно установила, что несколько слоёв графена, нагретые при температуре 300–400 °С в присутствии



порошкового хлорида железа ( $\text{FeCl}_3$ ) приводит к интеркаляции слоёв графена и хлорида железа. Электроны из хлорида железа увеличивают число носителей заряда в слоях графена, а результате чего поверхностное сопротивление слоя падает до 8,8 Ом на квадрат при видимой прозрачности материала 84 %. Новый материал имеет хорошую долговременную и температурную стабильность и во много раз лучше по характеристикам, чем сравнимые слои ИТО: при том же поверхностном сопротивлении последний имеет прозрачность лишь 75 %, а при той же прозрачности – сопротивление в 40 Ом на квадрат [5].

Датчики для диагностики заболеваний. В датчике используется графен, молекулы ДНК и флуоресцентные молекулы. Флуоресцентные молекулы соединяются с одиночной ДНК, которая в свою очередь связывается с графеном. Когда другая одиночная молекула ДНК связывается с ДНК, присоединенной к слою графена, и формируется двойная ДНК, которая свободно передвигается по графену, увеличивая уровень излучения (рисунок 4) [6].

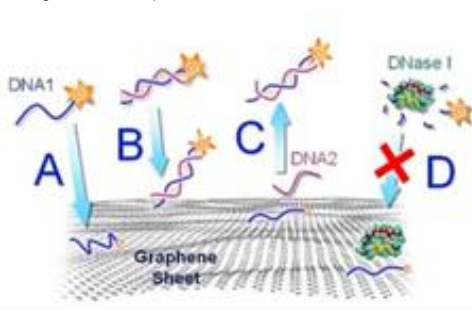


Рисунок 4 – Принцип распознавания поврежденных ДНК

## Литература

1. russianelectronics [Электронный ресурс] / Графеновый транзистор разогнали до 26 ГГц. – Режим доступа: <http://www.Russianelectronics.ru/developer-r/news/9318/doc/30562/> – Дата доступа: 19.04.2015

2. russianelectronics [Электронный ресурс] / Ученые создали первую в мире графеновую память. – Режим доступа: [Ошибка! Недопустимый объект гиперссылки. leader-r/news/company/2124/doc/31506/](http://www.Russianelectronics.ru/leader-r/news/company/2124/doc/31506/) – Дата доступа: 20.04.2015

3. russianelectronics [Электронный ресурс] / Новые возможности супер-конденсаторов с графеновыми электродами. – Режим доступа: <http://www.russianelectronics.ru/leader-r/news/9318/doc/58776/> – Дата доступа: 20.04.2015

4. russianelectronics [Электронный ресурс] / Графен: перспектива для ЖК-дисплеев. – Режим доступа: <http://www.russianelectronics.ru/developer-r/news/company/2125/doc/8396/> – Дата доступа: 21.04.2015

5. electronicsweekly [Электронный ресурс]/ Modified graphene beats ITO as see-through conductor. – Режим доступа: <http://www.electronic-weekly.com/news/research/materials-rd/modified-graphene-beats-ito-as-see-through-conductor-2012-05/> – Дата доступа: 22.04.2015

6. understandingnano [Электронный ресурс] / Graphene-DNA biosensor selective, simple to create – Режим доступа: <http://www.understanding-nano.com/graphene-dna-biosensor.html> – Дата доступа: 22.04.2015

**И.С. Мамичев (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **Е.М. Березовская**, канд. физ.-мат. наук, доцент

## **ОПРЕДЕЛЕНИЕ ПАРАМЕТРОВ ЭМПИРИЧЕСКОЙ ФОРМУЛЫ**

На сегодняшний день издано большое количество литературы по методике экспериментирования и обработки экспериментальных данных для построения эмпирических зависимостей. Научное творчество невозможно без умело поставленных экспериментов и грамотной обработки их результатов, которая позволяет извлечь из проделанной работы максимум формализованной, численно выраженной информации.

Однозначное определение эмпирической формулы дать нельзя, поскольку в разных областях науки она имеет разное значение и применение. К примеру, в экономике эмпирическая формула не выводится математически и не имеет особого смысла в научном понимании. Эта формула только приводит теоретические значения к эмпирическим, полученным опытным путем. Форму такой зависимости подбирает исследователь. Он же определяет конкретные значения параметров эмпирической формулы. В физике и математике эмпирической формулой называется математическое уравнение, полученное опытным путем, методом проб и ошибок или как приближенная формула из экспериментальных данных. Характерной особенностью таких формул является наличие эмпирических коэффициентов – специально подобранных параметров эмпирической формулы.

Рассматривались задачи о приближении функции, понятия аппроксимация и интерполяция, основы теории приближения функции для экспериментального получения зависимости, основные методы определения параметров эмпирической формулы. Разработано приложение для вычисления параметров эмпирической формулы. В данной работе рассматривалась аппроксимация эмпирической формулы по методу наименьших квадратов. Разработанная программа позволяет

анализировать данные, полученные экспериментальным путем, при помощи графического отображения на координатной плоскости. На основании анализа, можно определить вид эмпирической формулы (линейная или степенная функция). Далее, после определения вида формулы пользователем, программа определяет ее параметры. Программная часть работы была реализована в среде Borland Delphi 7.

**П.А. Марсиков (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **Г.Л. Карасёва**, канд. физ.-мат. наук, доцент

## **ИСПОЛЬЗОВАНИЕ СИСТЕМЫ ЭЛЕКТРОННОГО ДОКУМЕНТООБОРОТА В ИТ-КОМПАНИЯХ**

Семинары в ИТ-компаниях – традиционная форма повышения квалификации сотрудников. Семинары проводятся в форме коллективного обсуждения коллегами научной информации, навыков полученных опытным путем, наиболее актуальных методик и инструментов, применяемых в работе.

В зависимости от темпов роста компании рано или поздно возникает проблема подготовки проведения семинаров. При большом количестве сотрудников в компании возникает проблема своевременного оповещения персонала о предстоящих семинарах, затрудняется процесс регистрации на семинары. Возникает необходимость хранения информации о проведенных и предстоящих семинарах.

Решить данную проблему можно путем создания информационной системы, которая будет предназначена для автоматизации процесса подготовки проведения семинаров.

Для реализации такой задачи можно использовать платформу Lotus Notes. Данная технология предназначена для организации электронного документа оборота на предприятии, а также автоматизации совместной деятельности рабочих групп.

В отличие от привычного способа оповещения сотрудников, например, размещение сообщений на доске объявлений, платформа Lotus Notes позволяет разработать систему, которая будет оперативно оповещать сотрудников, зарегистрированных в системе, предоставлять удобный интерфейс регистрации на семинары, предоставлять удобную систему для хранения информации о проведенных и предстоящих семинарах.

Использование данной информационной системы позволяет сократить время, затрачиваемое на оповещение сотрудников, время

регистрации участников на семинар, а также получить гарантию, что все сотрудники будут своевременно проинформированы.

Система также позволяет автоматизировать процесс формирования групп участников семинара, создать механизм рассылки напоминаний о предстоящем семинаре, на который был зарегистрирован сотрудник. Также создать удобную инфраструктуру для хранения мультимедийных материалов, используемых в процессе проведения семинара.

Таким образом, автоматизировав, описанные выше процессы, IT-компания может решить проблему подготовки проведения семинаров, тем самым, освободив сотрудников от данного вида деятельности.

**А.А. Мартинович (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **Е.М. Березовская**, канд. физ.-мат. наук, доцент

## **ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ PHP. РАЗРАБОТКА WEB-САЙТА – «МИХИ»**

Данная работа посвящена одному из самых перспективных на сегодняшний день языков программирования – языку PHP. Технологию по имени PHP можно без преувеличения назвать революционной в индустрии разработки web-сайтов.

Для создания современных сайтов обычно используются: HTML, CSS, JavaScript (или VBScript), FLASH, PHP (или Perl) и реляционная база данных MySQL.

Клиентская среда (браузер, например Internet Explorer) является передним краем работы приложения. В этой среде отображаются HTML-страницы в окне и обслуживаются истории сеансов HTML-страниц, отображаемых в браузере в течение сессии. Объекты этой среды, следовательно, обязаны иметь возможность манипулировать страницами, окнами и историей. Для этого и нужен клиентский язык – JavaScript. При работе с сервером необходимы PHP (или Perl) и реляционная база данных MySQL.

PHP – это язык программирования, предназначенный для создания сайтов, позволяет автоматизировать работу с сайтом. Это скрипт-язык, встраиваемый в HTML, который интерпретируется и выполняется на сервере. Отличие PHP от JavaScript, состоит в том, что PHP-скрипт выполняется на сервере, а клиенту передается результат работы, тогда как в JavaScript-код полностью передается на клиентскую машину и только там выполняется.

В прошлом веке, чтобы создать сайт «на мировом уровне», достаточно было просто уметь работать с HTML и обладать художественным

вкусом. Спустя некоторое время требования усложнились: использование JavaScript и Dynamic HTML стало рутинным, а дизайн страниц, не имеющих подобных «украшений», считался устаревшим. Вскоре правилом «хорошего тона» для корпоративного сайта стало применение сложных программ на Perl или C++. Однако Web-дизайнеры не желали мириться с такой ситуацией. И на свет появился PHP – язык программирования, обладающий возможностями сложных скриптовых языков, но в то же время удивительно простой и легкий в изучении и применении.

Зачем же может потребоваться PHP? Разве обычных HTML и JavaScript недостаточно для создания красивой Web-страницы? Почему нужно выбирать PHP? PHP является внедряемым языком сценариев. Цель языка состоит в том, чтобы позволить веб-разработчикам быстро создавать динамически генерируемые страницы. Кроме того, разработчикам Web-приложений нет необходимости говорить, что web-страницы – это не только текст и картинки. Достойный внимания сайт должен поддерживать некоторый уровень интерактивности с пользователем: поиск информации, продажа продуктов, конференции и т. п. PHP работает как часть Web-сервера. В этом языке нет строгой типизации данных и нет необходимости в действиях по выделению / освобождению памяти. Программы, написанные на PHP, достаточно легко читаемы. Написанный PHP – код легко зрительно прочитать и понять.

Применение PHP можно разделить на две группы:

1. Самое простое (но очень мощное) применение – это использование PHP для создания сайтов из модулей. Записать все неизменяемые части сайта в отдельные файлы (меню, шапку, тело) и вместо того, чтобы писать одно и то же на каждой странице, просто вызывать вложение соответствующего модуля. Это уже значительно облегчит работу с сайтом. К примеру, если добавится новый пункт в меню.

2. Следующий этап – создание отдельных небольших программ. Это может быть голосование, гостевая книга, конференции «форум» и т. п. PHP применяется тогда, когда нужно какую-то информацию хранить на сервере. К примеру, гостевую книгу можно написать на JavaScript, но вот увидит её сообщения только тот, кто их добавлял.

В результате исследований был спроектирован и создан web-сайт «МИХИ». Web-сайт «МИХИ» – швейное интернет-ателье по пошиву и ремонту любого вида и любой сложности одежды с учётом особенностей фигуры и пожеланий клиента. Сайт состоит из клиентской и серверной части, все данные хранятся в базе данных на сервере в виде таблиц. Каждый пользователь имеет свой определенный доступ к данным, с определенными ограничениями, перед работой каждый

пользователь обязан зарегистрироваться. Данный сайт предназначен работать с клиентами, хранить информацию, изменять, удалять ее и добавлять. Разработка полностью носит функциональный характер и предназначена для полноценного использования.

**Н.П. Мельников (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Кулинченко**, ст. преподаватель

## **РАЗРАБОТКА ПРОЕКТА ОТОБРАЖЕНИЯ ДИНАМИЧЕСКОЙ ИНФОРМАЦИИ ПРИЕМНОЙ КОМИССИИ УНИВЕРСИТЕТА**

Приемная комиссия университета ведет статистику количества поданных заявлений на обучение по различным специальностям. Абитуриенты подают документы на обучение, в которых содержатся их личные данные, форма получения высшего образования (дневная или заочная), тип оплаты и суммарное количество баллов, с учетом результатов централизованного тестирования либо экзаменов на определенные специальности и среднего балла аттестата об общем среднем или среднем специальном образовании.

Условия и порядок приема абитуриентов на дневную и заочную формы получения высшего образования в учреждение образования «Гомельский государственный университет имени Франциска Скорины» определяются «Правилами приема в высшие учебные заведения», утвержденными Указом Президента Республики Беларусь от 07.02.2006 № 80 (с изменениями и дополнениями).

Существует план приема абитуриентов на специальности в котором отображены данные о количестве мест с оплатой за счет средств республиканского бюджета и мест на условиях оплаты студентом.

Статистика ведется подсчетом количества поданных документов в градации по суммарному количеству баллов, что и определяет общий конкурс. Отдельная статистика ведется для абитуриентов, поступающих на условиях целевой подготовки, без вступительных испытаний, вне конкурса.

Изучение структуры информационных потоков показало, что приемная комиссия университета использует электронную таблицу для хранения текущих данных о ходе вступительной кампании. Электронные таблицы (ЭТ) получили широкое применение в документообороте учреждений самых различных сфер деятельности и представляют собой удобный инструмент для автоматизации вычислений и обработки данных, представленных в виде двумерных массивов.

Также, в функции персонала, работающего в приемной комиссии, входит внесение входных данных, поступающих из документов абитуриентов, в ЭТ.

Основной задачей проекта стала разработка системы отображения, которая будет демонстрировать данные из ЭТ по всем специальностям факультетов в удобочитаемом виде на информационном дисплее в здании университета.

Алгоритм программы включает сканирование файла ЭТ параметризованным парсером, представление полученных данных в виде объектной модели и передачу объектов в качестве аргументов через программные интерфейсы на уровень представления. Сканирование файла происходит регулярно для предоставления актуальной информации. Уровень представления имеет слайдер для автоматической навигации по списку объектов и видеоплеер для проигрывания информационных роликов.

Настраиваемые параметры приложения включают: определение списка специальностей и имен других ключевых параметров для их поиска в ЭТ (такой подход увеличивает стабильность и улучшает масштабируемость приложения в случае изменения структуры исходного файла), интервал смены слайдов и пути к файлам на носителях информации.

Табличные процессоры предлагают необходимый и в большинстве случаев достаточный набор инструментов для анализа и обработки данных ЭТ пользователем. Однако, в данном случае необходимо, чтобы такой анализ и демонстрация результатов производились автоматически.

На этапе исследования подходящих технологий были найдены два решения для автоматизации процесса получения данных из файлов ЭТ Microsoft Excel. Таблицы Excel имеют два основных формата хранения – это файлы с расширением XLS, представленные бинарным BIFF файлом, упакованным в контейнер OLE, и файлы с расширением XLSX открытого формата Office Open XML, которые представляют собой архив, содержащий текстовые данные в формате XML, графику и другие вложения.

Для экспорта и миграции данных из файлов Microsoft Excel не зависимо от версии программной среды существуют API для различных языков программирования и сред выполнения. Так, для языков C++, VB, .NET доступен фреймворк ActiveX разрабатываемый компанией Microsoft. Однако официально управляющие элементы ActiveX поддерживаются только операционной системой Windows. Для кроссплатформенной среды исполнения Java существуют библиотеки,

разрабатываемые в рамках проекта Apache POI [1] и распространяемые с лицензией свободного программного обеспечения.

Первый способ решения задачи выполнен с помощью средств пакета Microsoft Office, языка программирования Visual Basic for Applications и фреймворка ActiveX. Была создана презентация Microsoft PowerPoint с поддержкой макросов в формате PPTM, и с помощью специальной вставки созданы объекты, связанные с исходным файлом ЭТ так, что изменения в исходном файле отображаются в презентации. После, в параметрах связей документа выбрано их автоматическое обновление, которое выполняется при открытии файла. В настройках демонстрации был выбран непрерывный цикл. И чтобы данные обновлялись динамически, добавлена функция, автоматически запускающая обновление связанных данных перед показом первого слайда. Так, стандартный метод `SlideShowWindow.View.CurrentShowPosition` определяет порядковый номер отображаемого слайда, а метод `ActivePresentation.Update Links` обновляет связи. Плюсами такого решения являются быстрая реализация и простое сопровождение, которое смогут осуществлять сами сотрудники приемной комиссии после изучения предоставленных инструкций. При этом решение полностью удовлетворяет поставленным требованиям.

Второе решение представлено веб-приложением, серверная часть которого написана на языке программирования Java и использует возможности библиотек Apache POI. Библиотеки POI предоставили возможность чтения файлов ЭТ XLS и XLSX через классы `HSSFWorkbook` и `XSSFWorkbook` соответственно, которые реализуют общий интерфейс `Workbook`. Это позволило сделать программный код независимым от формата источника и получать объекты `Sheet` (лист), `Row` (строка) и `Cell` (ячейка) из экземпляров типа `Workbook`. Объекты типа `Cell` и содержат значения ячеек ЭТ. Мощный бэк-энд приложения позволяет получать данные в виде объектной модели, что упрощает их дальнейшее использование. Архитектура приложения построена в соответствии с концепцией MVC. Недостатком этого решения становится более сложное сопровождение. Существенные изменения в структуре данных ЭТ могут потребовать изменений в исходном коде, перекомпиляции и повторного развертывания приложения на сервере. Преимущество в сравнении с первым решением – получение информации конечным пользователем через браузер на любом устройстве.

Информатизация становится определяющей сферой общественной жизни, ее рассматривают как один из основных ресурсов развития общества, а информационные системы и технологии как средство повышения производительности и эффективности человеческого труда.



Разработанная система позволяет абитуриентам получать информацию о ходе вступительной кампании в реальном времени.

### **Литература**

1. POI-HSSF and POI-XSSF – Java API To Access Microsoft Excel Format Files [Электронный ресурс]. – В режиме доступа: <http://poi.apache.org/spreadsheet/> – Дата доступа: 02.03.2015

**Н.П. Мельников (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Кулинченко**, ст. преподаватель

### **МИГРАЦИЯ ДАННЫХ ИЗ ЭЛЕКТРОННЫХ ТАБЛИЦ MICROSOFT EXCEL С ПОМОЩЬЮ СТОРОННИХ API**

Электронные таблицы (ЭТ) получили широкое применение в документообороте учреждений самых различных сфер деятельности и представляют собой удобный инструмент для автоматизации вычислений и обработки данных, представленных в виде двумерных массивов. Табличные процессоры предлагают необходимый и в большинстве случаев достаточный набор инструментов для анализа и обработки данных ЭТ пользователем. Однако, иногда необходимо, чтобы такой анализ проводился автоматически и максимально быстро.

Далее будет описано как удалось автоматизировать процесс получения данных из файлов ЭТ Microsoft Excel. Таблицы Excel имеют два основных формата хранения – это файлы с расширением XLS, представленные бинарным BIFF файлом, упакованным в контейнер OLE, и файлы с расширением XLSX открытого формата Office Open XML, которые представляют собой архив, содержащий текстовые данные в формате XML, графику и другие вложения.

Для экспорта и миграции данных из файлов Microsoft Excel не зависимо от версии программной среды существуют API для различных языков программирования и сред выполнения. Так, для языков C++, VB, .NET доступен фреймворк ActiveX, разрабатываемый компанией Microsoft. Однако, официально управляющие элементы ActiveX поддерживаются только операционной системой Windows. Поэтому выбор был сделан в пользу библиотек, разрабатываемых в рамках проекта Apache POI для кроссплатформенной среды исполнения Java и распространяемые с лицензией свободного программного обеспечения.

Библиотеки POI предоставили возможность чтения и создания файлов XLS и XLSX через классы HSSFWorkbook и XSSFWorkbook

соответственно, которые реализуют общий интерфейс Workbook. Это позволило сделать программный код независимым от формата источника и получать объекты Sheet (лист), Row (строка) и Cell (ячейка) из экземпляров типа Workbook. Объекты типа Cell и содержат значения ячеек ЭТ. Используя различные алгоритмы обработки, можно представить полученные данные в виде определенной модели и передавать через программные интерфейсы далее, на потребление конечным пользователям или хранение.

**А.В. Мельникова («ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **Д.С. Кузьменков**, канд. физ.-мат. наук, доцент

## **ИСПОЛЬЗОВАНИЕ СИСТЕМЫ МАТЛАВ ДЛЯ РЕШЕНИЯ ЗАДАЧ ТЕПЛОПРОВОДНОСТИ**

Разработка и исследование значительной части элементов современных сверхбольших интегральных схем связаны с решением задач математической физики, к которым относят задачи теплопроводности, диффузии, электростатики и электродинамики, задачи о течении жидкости, о распределении плотности электрического тока в проводящей среде, задачи о деформациях твердых тел и многие другие.

Подобные задачи описываются дифференциальными уравнениями в частных производных с дополнительными уравнениями, выражающими граничные и начальные условия. Нахождение точного аналитического решения, к сожалению, возможно лишь для весьма ограниченного круга одномерных задач при использовании целого ряда допущений. Для решения уравнений математической физики в случае нескольких измерений используют численные методы, позволяющие преобразовать дифференциальные уравнения или их системы в системы алгебраических уравнений. Точность решения определяется шагом координатной сетки, количеством итераций и разрядной сеткой компьютера.

Математическое описание задачи теплообмена включает систему дифференциальных уравнений в частных производных, описывающиеся нестационарными нелинейными уравнениями с граничными и начальными условиями, задающие взаимодействие объекта с окружающей средой [1]. В общем случае, это сложная для решения задача, если иметь в виду сложную геометрию реальных инженерных объектов, нелинейность уравнений, разнообразие и неустойчивость гидродинамических полей и т. п.

Универсальным средством решения таких задач становится все в большей мере численное моделирование на компьютерах. Во-первых,

численное моделирование востребовано в связи с усложнением энергетических технологий. Во-вторых, численное моделирование становится доступным благодаря появлению и усовершенствованию инженерных математических пакетов, одним из которых является MATLAB.

MATLAB – это высокопроизводительный язык для технических расчетов. Он включает в себя вычисления, визуализацию и программирование в удобной среде, где задачи и решения выражаются в форме, близкой к математической [2, 3]. Типичное использование MATLAB: математические вычисления, создание алгоритмов, моделирование, анализ данных, исследования и визуализация, научная и инженерная графика, разработка приложений, включая создание графического интерфейса.

MATLAB – это интерактивная система, в которой основным элементом данных является массив. Это позволяет решать различные задачи, связанные с техническими вычислениями, особенно в которых используются матрицы и вектора, в несколько раз быстрее, чем при написании программ с использованием «скалярных» языков программирования, таких как Си или Фортран.

MATLAB как язык программирования был разработан Кливом Моулером в конце 1970-х гг. Целью разработки служила задача дать студентам факультета возможность использования программных библиотек Linpack и EISPACK без необходимости изучения Фортрана. До сих пор в Интернете можно найти версию 1982 г., написанную на Фортране, распространяемую с открытым исходным кодом. Первоначально MATLAB предназначался для проектирования систем управления, но очень быстро завоевал популярность во многих других научных и инженерных областях. Он также широко использовался и в образовании, в частности, для преподавания линейной алгебры и численных методов.

Задача теплопроводности была реализована в виде m-файла, не содержащего функций, созданных пользователем. Решение исследуемой задачи теплопроводности сводится к решению системы обыкновенных дифференциальных уравнений. Для получения требуемых коэффициентов в среде MATLAB решается задача квадратичного программирования. В программе используются встроенные библиотеки MATLAB для решения обыкновенных дифференциальных уравнений, а также задач оптимизации.

## Литература

1. Борухов, В.Т. Отслеживание заданной программы взвешенных температур и восстановление коэффициентов теплообмена /

В.Т. Борухов, О.И. Костюкова, М.А. Курдина // Инженерно-физический журнал НАН Беларуси. – 2010. – Том 83, № 3. – С. 584–592.

2. Говорухин, В. Компьютер в математическом исследовании / В. Говорухин, Б. Цибулин. – Спб.: Питер. – 2001. – 624 с.

3. Васильев, В.В. Математическое и компьютерное моделирование в среде MATLAB / В.В. Васильев, Л.А. Симак, А.М. Рыбникова. – Киев: «АкадемПресс». – 2008. – 349 с.

**Е.С. Мельченко (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **Н.А. Шаповалова**, ст. преподаватель

## **ПРОЕКТ АВТОМАТИЗАЦИИ УЧЕТА РАБОТЫ СЕРВИСНОГО ЦЕНТРА**

В основе данного проекта лежит задача создания программного комплекса, охватывающего весь перечень задач, с которыми постоянно сталкиваются сотрудники сервисного центра. Это организация, в рамках которой предлагаются услуги широкополосного доступа в качественный и надежный Интернет, доступный абонентам во всех городах Республики Беларусь, доступ сети по проводным и беспроводным технологиям, а так же предоставление технического обслуживания оборудования для доступа в сеть Интернет.

Проведя анализ существующих на сегодняшний день программных средств, было решено, что для автоматизации такого разностороннего перечня задач лучше всего подойдет программный продукт Microsoft Access 2010. Встроенные прикладные объекты данного программного продукта отлично подходят для поставленной задачи.

База данных сервисного центра содержит весь перечень документов и отчетов, с которыми сталкиваются сотрудники сервисного центра.

В ходе выполнения проекта было решено множество проблем, связанных с созданием документов, таблиц, запросов и отчетов, необходимых для быстрого доступа к необходимой информации. Был спроектирован и реализован весь перечень отчетов, с которыми постоянно сталкиваются сотрудники сервисного центра. Все отчеты имеют параметры, что позволяет формировать их за любой промежуток времени.

Для удобства пользователя был реализован удобный, наглядный и интуитивно понятный интерфейс.

Для запуска приложения пользователю понадобится пройти аутентификацию. Это позволит уменьшить количество рисков, связанных

с искажением и кражей корпоративных данных злоумышленниками. Причем, все пользователи имеют различный спектр возможностей по манипуляции над данными. По полномочиям сотрудников за каждым пользователем закреплены определенные роли на создание, чтение, редактирование и удаление данных из базы.

Результатом данной работы является создание полноценной, рабочей базы данных «Сервисный центр». Данная база полностью функционирует и готова к использованию. Разработанный интерфейс позволяют свободно обрабатывать необходимый спектр данных. Программный комплекс правильно выполняет запросы пользователя и соответствует предъявляемым требованиям.

**П.А. Меньшаков** (УО «ГГТУ им. П.О. Сухого», Гомель)  
Науч. рук. **И.А. Мурашко**, д-р техн. наук, профессор

## **СРЕДСТВО ГОЛОСОВОЙ ИДЕНТИФИКАЦИИ ПОЛЬЗОВАТЕЛЯ**

В настоящее время, голосовая идентификация, как и биометрия в целом, уже получила широкое распространение. Самый простой пример – сканеры отпечатков пальцев, установленные почти на каждом ноутбуке.

В области информационных технологий биометрические данные используются в качестве формы управления идентификаторами доступа и контроля доступа. Также биометрический анализ используется для выявления людей, которые находятся под наблюдением [1]. Так же биометрия предусматривает и поведенческий анализ объекта. К ним относятся ходьба, жесты и т. п.

Процесс авторизации, при использовании биометрии, довольно прост. При помощи устройства, предназначенного для получения той или иной характеристики, сканируются текущие данные идентифицируемого и сравниваются с полученными ранее данными. Биометрические системы в отличие от бумажных идентификаторов (паспорт, водительские права, удостоверение личности), от пароля или персонального идентификационного номера (ПИН), биометрические характеристики не могут быть подвергнуты краже, не могут быть утеряны или забыты. Довольно длительное время отпечатки пальцев используются для идентификации преступников и предотвращения воровства или мошенничества. Некоторые люди умеют имитировать голоса, но, это требует особых навыков, которые не часто повстречаешь в обыденной жизни [2].

При помощи микрофона получается запись голоса идентифицируемого и отправляется на ЭВМ. Наиболее оптимальным является получение WAV файла, в виду простоты работы с ним.

Полученную запись голоса необходимо разделить на кадры. Разделение на кадры представлено на рисунке 1. Данное действие необходимо для более простой работы с записанной звуковой дорожкой.

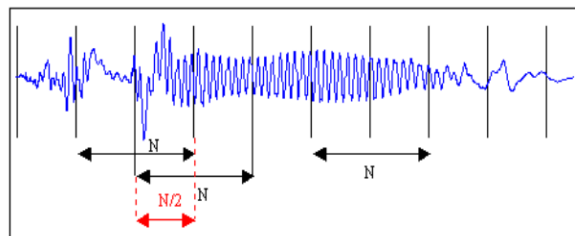


Рисунок 1 – График звуковой волны

Следующим этапом является устранение нежелательных эффектов и шумов. Это необходимо для того, чтобы записи, полученные в разное время соответствовали друг другу независимо от сторонних факторов. Существует множество способов, при помощи которых можно уменьшить шумовые эффекты. Мною использовалось умножение каждого кадра на особую весовую функцию «Окно Хемминга»:

$$\omega(n) = 0,53836 - 046164 * \cos\left(\frac{2\pi n}{N-1}\right),$$

где  $n$  – порядковый номер элемента в кадре, для которого вычисляется новое значение амплитуды,

$N$  – длина кадра (количество значений сигнала, измеренных за период).

Полученные кадры преобразуется в их частотную характеристику при помощи прогонки через «Быстрое Преобразование Фурье»:

$$X_k = \sum_{i=0}^{N-1} x_n e^{-\frac{2\pi i}{N} kn},$$

где  $N$  – длина кадра (количество значений сигнала, измеренных за период),

$x_n$  – амплитуда  $n$ -го сигнала,

$X_k$  –  $N$ -комплексных амплитуд синусоидальных сигналов, слагающих исходный сигнал.

На сегодняшний день наиболее успешными являются системы распознавания голоса, использующие знания об устройстве слухового

аппарата. Они базируются на том, что ухо интерпретирует звуки не линейно, а в логарифмическом масштабе. В виду данных особенностей необходимо привести частотную характеристику каждого кадра к «мелам». Зависимость представлена на рисунке 2.

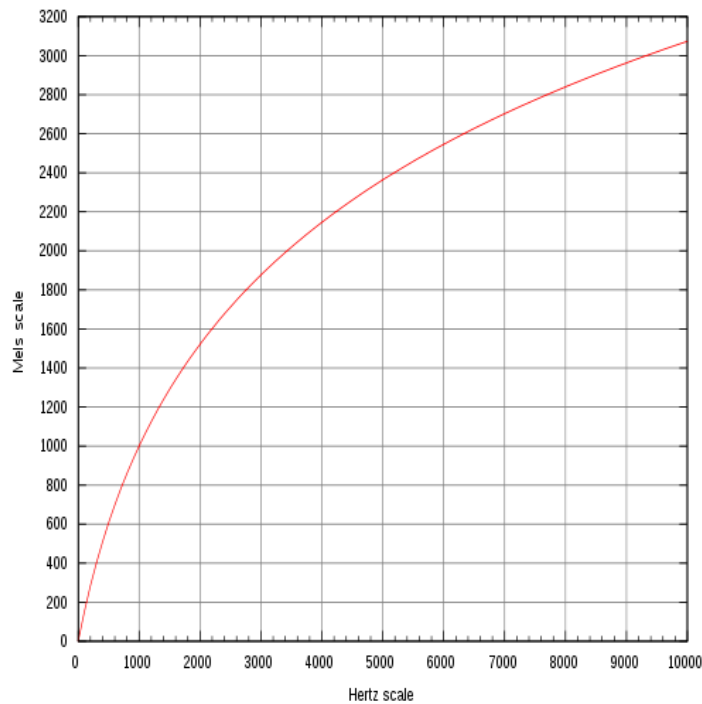


Рисунок 2 – График зависимости высоты звука (в мелах) от его частоты

Для перехода к «мел» характеристике используется следующая зависимость:

$$m = 1127 \log_e \left( 1 + \frac{f}{700} \right),$$

где  $m$  – частота в мелах,  
 $f$  – частота в герцах.

Это последнее действие, необходимое для последующего преобразование в вектор характеристики, который, впоследствии, сравнивается с базой голосовых записей. Вектор будет состоять из мел-кепстральных коэффициентов, получить которые можно по следующей формуле:

$$C_n = \sum_{k=1}^K (\log S_k) \left[ n \left( k - \frac{1}{2} \right) \frac{\pi}{K} \right],$$

где  $C_n$  – мел-кепстральный коэффициент под номером  $n$ ,  
 $S_k$  – амплитуда  $k$ -го значения в кадре в мелах,

$K$  – наперед заданное количество мел-кепстральных коэффициентов  $n \in [1, K]$ .

Полученный вектор характеристик добавляется в базу данных, для последующего сравнения с ним.

Однако более оптимальным вариантом является использование нескольких записей одного и того же голоса. Заранее определенное количество образцов голоса можно использовать для обучения нейронной сети.

### Литература

1. [Bosi M., 2003] Introduction to digital audio coding and standards / M. Bosi, R.E. Goldberg-Springer Science+Business, Media USA. – 2003. – 434 p.
2. [You Y., 2010] AudioCoding: Theory and Applications / Y. You NY: Springer, 2010. – 349 p.
3. [Загуменнов, А. П., 1999] Компьютерная обработка звука / А. П. Загуменнов. – М.: ДМК, 1999. – 384 с.

**Д.В. Мигун (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **П.Л. Чечет**, канд. техн. наук, доцент

### УСТРОЙСТВА ОТОБРАЖЕНИЯ ИНФОРМАЦИИ С МЕХАНИЧЕСКОЙ РАЗВЕРТКОЙ

Бегущая строка – электронное устройство, предназначенное для отображения текстовой и графической информации. В самом простом варианте текст просто «бежит», двигаясь справа налево. Отображающая матрица обычно состоит из светодиодов. Часто помимо простого текста, бегущие строки могут генерировать простейшую графику и накладывать различные эффекты на отображаемый текст.

Перед проектированием устройства отображения информации с механической развёрткой, нужно учесть, что оно может использоваться не только в помещениях, но и на улице. Агрессивная атмосфера и суровый климат вынуждают производителей использовать пылевлагозащищённые корпуса, проклеивать печатные платы специальным пластиком, утеплять конструкцию и предусматривать вентиляцию. Варианты в северном исполнении имеют автоматический подогрев. Но помимо тяжёлых условий имеет место ещё один фактор – солнечные лучи, которые засвечивают символы и снижают контрастность изображения. Для того чтобы противостоять влиянию засветки,



фирмы-производители используют более яркие светодиоды: если для помещений яркость каждого светодиода до 1–2 Кд, то для уличного варианта этот показатель выше.

«Бегущая строка» зачастую используется в рекламных целях или для привлечения внимания рекламными агентствами, сетями магазинов, банками, спортивными сооружениями и многими другими, кто хоть как-то взаимодействует с массами людей. Также табло «Бегущая строка» может использоваться и в некоммерческих целях, например, для визуального оповещения населения, сообщения новостей и оперативной информации, объявления остановок в общественном транспорте, отображения текущего времени, даты, температуры окружающего воздуха и т. п.

### **Литература**

1. Ефимчик, М.К. Технические средства электронных систем : учеб. пособие / М.К. Ефимчик. – Мн.: Тесей, 2006. – 304 с.
2. Огарева, Н.П. Общая электротехника и электроника : учеб. пособие / Н.П. Огарева. – М.: Мордовский университет, 2002. – 290 с.

**Д.В. Мигун (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **П.Л. Чечет**, канд. техн. наук, доцент

### **МЕХАНИЧЕСКАЯ РАЗВЁРТКА ДЛЯ ОТОБРАЖЕНИЯ ИНФОРМАЦИИ**

Работы по применению механической развёртки при отображении информации активно велись около ста лет назад, когда уровень развития техники и электроники не позволял использовать другие способы отображения информации. В настоящее время применение механической развёртки позволяет создавать интересные запоминающиеся конструкции, выделяющиеся из общей массы имеющихся устройств отображения наличием двигающихся частей. Современный уровень развития элементной базы во многом позволяет с минимальными затратами разрабатывать и создавать устройства с механической развёрткой для отображения информации.

Наиболее известное применение механической развёртки любителями – это часы, которые известны как «часы–пропеллер» или «виртуальные часы». Их особенность состоит в том, что изображение часов создается быстро вращающейся линейкой светодиодов. При взгляде на эти часы создается иллюзия, будто они проецируются в воздухе!

Часы с механической разверткой, Propeller clock, часы Боба Блика – это устройство имеет много названий, но в основу его работы положена разработка 80-х гг. позапрошлого столетия (1884 г.) предложенная немецким техником и изобретателем Паулем Нипковым. Это устройство получило название в честь своего изобретателя – диск Нипкова и послужило основой механического телевидения, которое просуществовало вплоть до 1939 г. и было широко распространено в Германии. Диск Нипкова имеет ряд отверстий расположенных по спирали, за диском располагается источник освещения, яркость которого модулируется видеосигналом.

Предлагаемое табло с помощью небольшого числа светодиодов создает относительно сложные графические изображения, для которых при обычном способе формирования потребовались бы сотни светодиодов. Как в кино или на телевидении, здесь используется инерционность человеческого зрения. Светодиоды, расположенные на вращающейся линейке, вспыхивают в определенном порядке. При частоте вращения около 25 Гц создается иллюзия, что «картинка» висит в воздухе.

Прототипами устройства послужили неоднократно описанные в Интернет публикациях «пропеллерные часы». Создаваемое подобным табло изображение может быть как статичным, так и несложной анимацией (медленно вращающаяся фигура, изменение числа ее элементов). Если использовать светодиоды повышенной яркости, изображение будет хорошо различимо не только в сумерках, но и днем в отсутствие, конечно, прямого солнечного света. Объем памяти примененного в устройстве МК позволяет одновременно хранить в нем информацию, достаточную для формирования пяти-шести разных фигур.

Главный недостаток табло – наличие в нем вращающегося узла – заставляет позаботиться о защите зрителей от случайных травм. Табло необходимо устанавливать за прозрачным экраном, в витрине, или подвешивать его на недоступной зрителям высоте.

Табло изготовлено из компьютерного вентилятора, на роторе которого закреплен вращающийся узел с блоком управления и двумя линейками светодиодов – красных и зеленых. Напряжение питания на этот узел поступает через высокочастотный трансформатор, вторичная обмотка которого размещена на роторе и вращается относительно первичной, укрепленной на неподвижном статоре. Генератор подаваемых на первичную обмотку высоко частотных импульсов и стабилизатор частоты вращения двигателя собраны на отдельной плате.

Бегущая строка – электронное устройство, предназначенное для отображения текстовой и графической информации. В самом простом

варианте текст просто «бежит», двигаясь, справа налево. Отображающая матрица обычно состоит из светодиодов. Очень часто помимо простого текста, бегущие строки могут генерировать простейшую графику и накладывать различные эффекты на отображаемый текст.

Светодиодные табло «бегущая строка» могут использоваться не только в помещениях, но и на улице. Агрессивная атмосфера и суровый климат вынуждают производителей использовать пылевлагозащищённые корпуса, проклеивать печатные платы специальным пластиком, утеплять конструкцию и предусматривать вентиляцию. Варианты в северном исполнении имеют автоматический подогрев. Но помимо тяжёлых условий имеет место ещё один фактор – солнечные лучи, которые засвечивают символы и снижают контрастность изображения. Для того, чтобы противостоять влиянию засветки, фирмы-производители используют более яркие светодиоды: если для помещений яркость каждого светодиода до 1–2 Кд, то для уличного варианта этот показатель гораздо больше.

«Бегущая строка» зачастую используется в рекламных целях или для привлечения внимания рекламными агентствами, сетями магазинов, банками, спортивными сооружениями и многими другими, кто хоть как-то взаимодействует с массами людей. Также табло «Бегущая строка» может использоваться и в некоммерческих целях, например, для визуального оповещения населения об опасности органами ГО и ЧС, сообщения новостей и оперативной информации, объявления остановок в общественном транспорте, отображения текущего времени, даты, температуры окружающего воздуха и т. п.

Монохромные и полноцветные бегущие строки. Светодиодный экран – это устройство, позволяющее отображать текстовую и графическую информацию. Главным его преимуществом является динамичность – возможность демонстрировать большое количество информации на сравнительно небольшой рекламной площади табло, используя для этого множество визуальных эффектов.

Предприятия все чаще отдают предпочтение LED вывескам для повышения продаж по ряду причин:

- светодиодная бегущая строка выделяет фирму среди конкурентов;
- светящееся анимационное табло может быть установлено в различных местах: в окне магазина, под козырьком, внутри здания;
- благодаря способностям светодиодов вывеска заметна днем и ночью на больших расстояниях. Таким образом, оборудование выполняет свою работу круглосуточно;
- светящееся LED табло даже минимальных размеров способно уместить обширный объем актуальной и полезной информации;

– электронная бегущая строка универсальна для организаций любого рода занятий. Светодиодная наружная реклама устанавливается в зданиях больших торговых центров, банках, административных учреждениях, развлекательных заведениях, открытых спортивных площадках;

– светодиодные строки вобрали в себя все передовые технологии, позволяющие воспроизводить любые световые эффекты, любую анимацию и видео;

– на таких светодиодных вывесках возможно отобразить рисунки, символы, изображения, в том числе, полноцветные корпоративные логотипы и видео.

Светодиодная бегущая строка – лучшее решение для повышения интереса покупателей. Минимальное время, за которое конструкция окупает себя, составляет менее месяца.

**К.О. Мисоченко (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Д. Левчук**, канд. техн. наук, доцент

## **РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ ОБРАБОТКИ ДАННЫХ ПО РЕГРЕССИОННОМУ ТЕСТИРОВАНИЮ**

Большинство компаний, которые занимаются разработкой и тестированием программного обеспечения, используют регрессионное тестирование как одно из главных для обеспечения контроля качества.

Регрессионное тестирование проводится в различных случаях: после изменения существующего функционала, после добавления нового, после изменений в связанных системах. В любом случае, основное назначение регрессионного тестирования – убедиться, что разработанный ранее функционал по-прежнему работает корректно. Тесты, включаемые в регрессионный набор, могут быть как функциональными, так и не функциональными. Как правило, целесообразной с точки зрения затрат и сроков бывает автоматизация регрессионного набора.

Для того чтобы обрабатывать, сравнивать и получать результаты о всех итерациях тестирования было создано приложение, которое автоматически собирает все необходимые данные.

Для разработки использовался MEAN стек, который включает в себя:

– документно-ориентированную базу данных MongoDB,

– фреймворк Express для эффективной и наглядной работы с серверной частью приложения,

– Angular.js как один из главных инструментов для разработки клиентской части приложения,

– Node.js – программная платформа, построенная на Chrome V8 Javascript движке, позволяющая легко строить быстрые, масштабируемые сетевые приложения.

MEAN является Full-stack фреймворком, так как позволяет использовать на сервере и клиенте один язык программирования – JavaScript. Приложение позволяет собирать полную статистику по отдельной итерации в регрессионном тестировании, а именно: количество тестов, количество успешных/проваленных/пропущенных тестов, номер сборки, время начала и завершения отдельной итерации/теста/метода, а так же их статусы. Есть возможность сравнивать разные итерации между собой по различным критериям и создавать PDF и HTML отчеты по отдельным итерациям или же по их сравнению. Дополнительными данными могут быть файлы с событиями в проваленных тестах, для их дальнейшего изучения и поиска решения.

Разработанное приложение значительно облегчит обработку информации о регрессионном прогоне, а также просмотр разницы между различными прогонами тестов в регрессии.

**К.О. Мисоченко (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Д. Левчук**, канд. техн. наук, доцент

## **АВТОМАТИЗАЦИЯ РУТИННЫХ ЗАДАЧ НА ПРИМЕРЕ РАЗРАБОТКИ ПРИЛОЖЕНИЯ ДЛЯ ПОСТРОЕНИЯ ОТЧЕТОВ ПО СТАТИСТИКЕ РЕГРЕССИОННОГО ТЕСТИРОВАНИЯ**

В процессе написания приложения многие разработчики сталкиваются с выполнением скучных и рутинных, но от того не менее важных задач, которые приходится постоянно выполнять в процессе разработки проекта. Такими задачами могут быть запуск модальных тестов, минификация JavaScript файла, конкатенация файлов, препроцессинг CSS, деплой файлов на конечный сервер и многие другие. Такие задачи приходится выполнять группами и по многу раз на день в активной стадии написания проекта. По-этому было решено все задачи автоматизировать при помощи таск-менеджера gulp.js. Это довольно простая утилита, которая позволяет сэкономить очень много времени и помогает сохранять фокус на задачах, связанных непосредственно с разработкой проекта.

Для работы с gulp.js нужно создать конфигурационный файл, где и прописываются задачи, которые нужно автоматизировать. Конкретная задача решается одним плагином. Каждый плагин выполняет только одно простое действие, это ключевая мысль gulp. Здесь нет плагинов, дублирующих действия друг друга и конфликтующих между собой и основным функционалом.

В проекте были задействованы следующие плагины:

- gulp-cssmin, минификатор для css файлов;
- gulp-uglify, минификатор для js файлов;
- gulp-uncss, удаление неиспользуемого CSS;
- gulp-imagemin, оптимизация изображений;
- gulp-rename, переименование файлов.

Для достижения наиболее полной отдачи от автоматизации рутинных задач было принято решение один раз настроить таск-менеджер и не беспокоится каждый раз о запуске задач. Для этого был использован плагин gulp-watch, который отслеживает изменения файлов и каких-либо других действий разработчика и запускает задачи автоматически.

Gulp – по-настоящему быстрый инструмент, построенный на Node.js, очень сильно упрощающий жизнь разработчика. Он легко настраивается и расширяется. Большинство готовых расширений устанавливаются одной командой вместе со всеми зависимостями.

**А.В. Михно (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **Н.А. Шаповалова**, ст. преподаватель

## **РАЗРАБОТКА МУЛЬТИМЕДИЙНОГО РЕСУРСА ПО ДИСЦИПЛИНЕ «ОРГАНИЗАЦИЯ ПРОИЗВОДСТВА И УПРАВЛЕНИЕ ПРЕДПРИЯТИЕМ»**

Целью дипломного проектирования является создание мультимедийного обучающего ресурса, позволяющего повысить эффективность процесса обучения студентов по дисциплине «Организация производства и управление предприятием». Данный ресурс состоит из двух частей. В первой части представлен лекционный материал, вторая часть содержит тесты, которые позволят обучающемуся контролировать степень усвоения материала.

Для создания обучающего ресурса была выбрана среда разработки moodle. Moodle – это инструментальная среда для разработки как отдельных онлайн-курсов, так и образовательных веб-сайтов. В основу

проекта положена теория социального конструктивизма и ее использование для обучения. Moodle распространяется свободно, как программное обеспечение с открытым исходным кодом. Система Moodle прошла сертификацию соответствия стандарту SCORM 1.2, поэтому на ее основе можно создавать полноценные учебные ресурсы для дистанционного обучения.

В ходе проектирования были рассмотрены существующие стандарты, особенно SCORM – набор спецификаций и стандартов, которые представлены разными организациями. Они все сгруппированы в три основных категории: модель объединения содержания («Content Aggregation Model (CAM)»), средства управления работой программы («Run-Time Environment (RTE)») и последовательность и навигация («Sequencing and Navigation (SN)») (представлена в SCORM 2004).

Для того чтобы более точно понять, как должна работать система, все чаще используется описание функциональности системы через варианты использования (Use Case или прецеденты). Варианты использования это – описание последовательности действий, которые может осуществлять система в ответ на внешние воздействия пользователей или других программных систем. Варианты использования отражают функциональность системы с точки зрения получения значимого результата для пользователя, поэтому они точнее позволяют ранжировать функции по значимости получаемого результата.

## Литература

1. Свободная энциклопедия Википедия [Электронный ресурс]. – 2015. – Режим доступа: <http://ru.wikipedia.org/>. – Дата доступа: 15.04.2015.
2. Официальный сайт разработки moodle [Электронный ресурс]. – 2015. – Режим доступа: <https://moodle.org/>. – Дата доступа: 15.04.2015.

**А.С. Мкртычян (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Леванцов**, ст. преподаватель

## НАКОПИТЕЛИ ИНФОРМАЦИИ НА ОСНОВЕ ДНК

На данный момент актуальным является вопрос хранения больших объемов информации. Последней разработкой в данной области является применение свойств ДНК для структурирования хранения информации.

Основной принцип использования этой технологии заключается в возможности синтеза лент ДНК и считывания из нее.

Реализуется данный принцип следующим образом: ноль кодируют нуклеотидами аденином или цитозином (А и С), единицу гуанином или тимином (G и T), после чего собирают двойную спираль нужной длины.

Однако у данной технологии имеются ограничения – спираль ДНК не может быть бесконечно длинной. Это является следствием естественных ограничений и несовершенства используемой техники. По этой причине «файл» разбивают на отрезки, вмещающие примерно сто байт каждый. Для чтения лента ДНК так же разрезается, поэтому основная проблема – это пометить отрезки, чтобы после чтения их можно было собрать в первоначальном порядке. Принимая во внимание механическую, термическую, химическую хрупкость ДНК, необходимо используются дополнительные средства коррекции ошибок. Отрезки помечаются порядковыми номерами и подвергаются помехоустойчивому кодированию.

Последним достижением в области обеспечения защиты лент ДНК от неблагоприятного воздействия является разработка швейцарских ученых: отрезки ДНК-спирали обёртываются в нанометровых размеров защитную оболочку из кварцевого стекла. С помощью этого возможно сохранив на ДНК 83 килобайта данных, прочесть их спустя неделю хранения при температуре 70 градусов Цельсия. При пониженных температурах данные на ДНК накопителе могут сохраняться на протяжении большого количества лет.

Недостатком технологии является большая стоимость процедур записи-чтения и невозможность организации доступа к произвольной ячейке.

Таким образом, несмотря на уязвимость при длительном нахождении при высоких температурах, создание накопителей на основе ДНК позволит решить проблему надежного хранения больших объемов информации.

**В.В. Муха (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Кулинченко**, ст. преподаватель

## **ЭНЕРГОЭФФЕКТИВНАЯ ТОПОЛОГИЧЕСКАЯ ПАМЯТЬ TRAM**

Группа, в которую вошли исследователи из японского Национального института науки и передовых технологий (National Institute of Advanced Industrial Science and Technology, AIST), университета Нагойи (Nagoya University) и ассоциации Low-power Electronics Association & Project (LEAP), создала опытные образцы топологической памяти



TRAM (topological-switching RAM), нового типа памяти на основе фазовых переходов, изготовленной на основе кристаллической решетки сложного сплава GeTe/Sb<sub>2</sub>Te<sub>3</sub>. Образцы памяти нового типа продемонстрировали крайне высокие показатели энергетической эффективности, что может привести к появлению высокоскоростных устройств хранения данных, таких, как SSD-диски, потребляющих при своей работе совсем незначительное количество энергии.

Память на основе фазовых переходов работает за счет перехода материала активного участка ячейки памяти из кристаллического в аморфное состояние и наоборот под воздействием электрического тока с определенными параметрами. Фазовые переходы в кристаллической решетке показаны на рисунке 1.

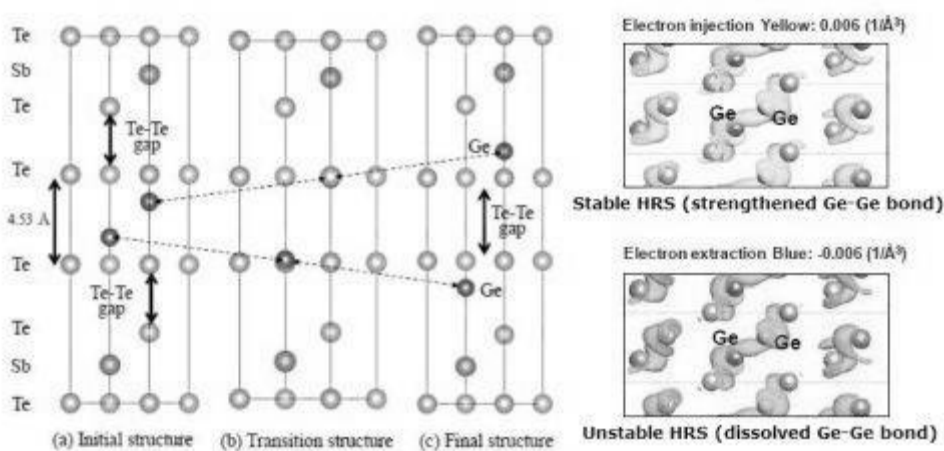


Рисунок 1 – Фазовые переходы в кристаллической решетке

В топологической памяти TRAM сопротивление активного участка ячейки памяти происходит за счет перемещения в кристаллической решетке атомов германия (Ge) на небольшое расстояние. Идея создания памяти подобного типа была обнародована представителями ассоциации LEAP в декабре 2013 г., а уже в июне 2014 г. при помощи обычной CMOS-технологии были созданы первые образцы TRAM-памяти, которая продемонстрировала превосходные энергетические показатели по сравнению с другими типами памяти на основе фазовых переходов.

Используя первый опытный образец TRAM-памяти, и проведя анализ структуры его кристаллической решетки при помощи электронного микроскопа, исследователи оптимизировали структуру ячеек памяти, что позволило улучшить ряд их характеристик по сравнению с первыми образцами. Оптимизированная структура решетки показана на рисунке 2.

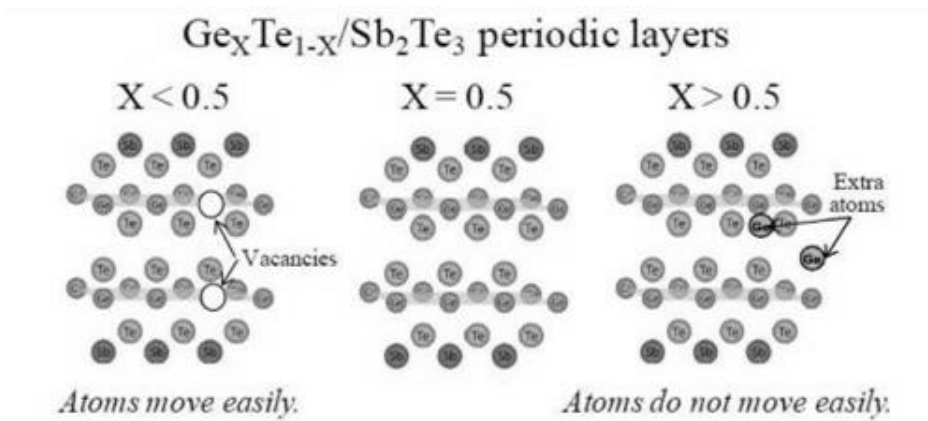


Рисунок 2 – Оптимизированная структура решетки

Ключом этих изменений стала новая структура кристаллической решетки, основанная на расчетной модели перемещения атомов германия, которая требует незначительного количества энергии для перемещения этих атомов. Структура кристаллической решётки под микроскопом показана на рисунке 3.



Рисунок 3 – Структура решетки на атомном уровне

Такой подход позволил снизить рабочее напряжение до 0,7 Вольт, на 30 процентов, и рабочий ток до 55 мкА, на 50 процентов, по сравнению с аналогичными характеристиками ячеек TRAM-памяти предыдущего поколения. График рабочего тока показан на рисунке 4.

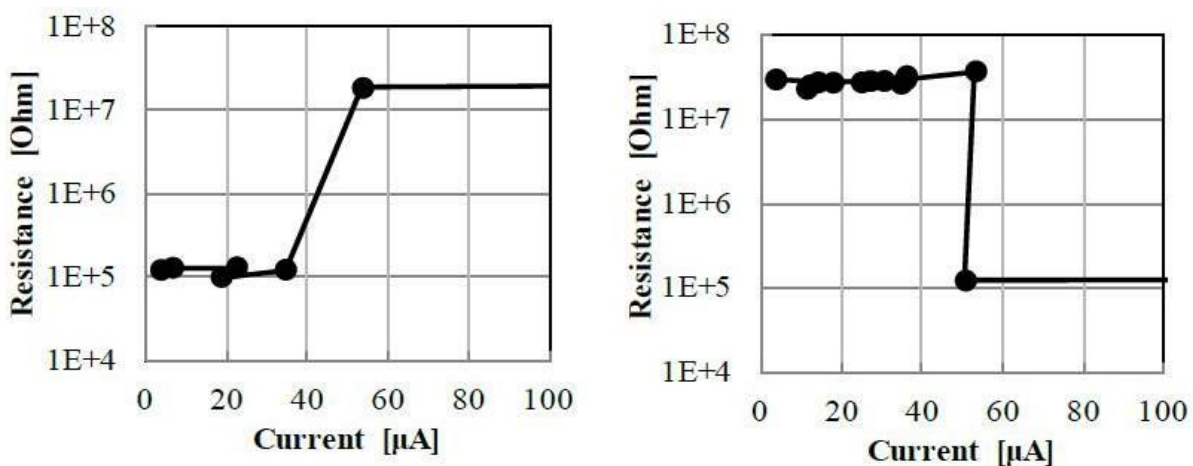


Рисунок 4 – Рабочий ток предыдущего поколения и оптимизированного

Кроме этого в 4 раза было уменьшено значение тока стирания и записи, что позволило существенно поднять скорость записи и снизить расход энергии в момент записи и стирания.

Исследователи изготовили несколько различных вариантов ячеек TRAM-памяти, используя наборы материалов, имеющих разные показатели удельной теплопроводности. Измерения характеристик изготовленных ячеек показало, что рабочее напряжение этих ячеек практически не зависит от используемых материалов, в отличие от других типов памяти на фазовых переходах, рабочее напряжение которых сильно зависит от удельной теплопроводности их материалов.

Специалисты ассоциации LEAP полагают, что устройства хранения на основе TRAM-памяти найдут применение в первую очередь в информационных центрах большого масштаба. Этому максимально будет способствовать то, что скорость работы TRAM-памяти почти в 100 раз превышает скорость работы самых лучших образцов современной NAND flash-памяти, и, благодаря этому, один TRAM-чип по объему и производительности сможет заменить до 64 чипов обычной flash-памяти.

**Н.А. Навныко (УО «ГГУ им Ф. Скорины», Гомель)**

Науч. рук. **В.Д. Левчук**, канд. техн. наук, доцент

## **РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ РЕКРУТИНГА СОТРУДНИКОВ НА DJANGO FRAMEWORK**

Данное веб-приложение предназначено для облегчения поиска и рекрутинга сотрудников клиентами сайта. Первая группа клиентов – компании, которые регистрируется на сайте, после чего размещают вакансии, публикуют новости и события, связанные с компанией, размещают контактные данные компании, результаты работы. Вторая группа клиентов – люди в поиске работы, которые могут после регистрации просматривать вакансии компаний, или сотрудники этих компаний, которые могут оформлять подписку на новости и события компании, а также самостоятельно организовывать мероприятия и публиковать новости, связанные или несвязанные с деятельностью компании.

Для разработки базы данных была использована система управления базами данных (СУБД) PostgreSQL. PostgreSQL – свободная объектно-реляционная СУБД.

Для описания структуры базы данных в Django используется понятие модели. Модель в Django – это описание данных, которые хранятся

в базе данных, выполненное в виде кода на языке Python. Django использует модель для фонового выполнения SQL и возвращает удобные структуры Python с данными, представляющими записи в таблицах базы данных. Все модели описываются в каждом приложении в файле `models.py`. Обычно каждая модель соответствует одной таблице в базе данных и, обычно, каждый атрибут модели соответствует столбцу в таблице. Имя атрибута модели соответствует имени столбца таблицы, а тип атрибута соответствует типу столбца.

Для основы приложения нам необходим минимальный набор моделей. Модель Пользователь в Django создается автоматически. Данная модель необходима для авторизации пользователей в системе. Для описания клиентов-компаний нам необходимы следующие модели: модели Группа компания и Офис компании, Новости, Работа (для описания результатов работы), Вакансии, События.

Веб-приложение подразделяется на две составные части: административную и основную. Административная часть в Django Framework создается автоматически (если проект был создан командой `python manage.py startproject`). Административная часть приложения предназначена для управления данными. При регистрации модели администратору сайта доступен список объектов этой модели, которые он может редактировать или создавать новые объекты. Административная часть в Django Framework может быть изменена под любые нужды. В текущем приложении она используется для управления данными из базы данных, назначения прав пользователям.

Основная часть приложения Django создается с помощью представлений (для описания контроллеров) и шаблонов (для создания страниц сайта). В Django шаблоны принято наследовать от базового. В текущем приложении базовый шаблон включает в себя основное меню сайта и баннеры (`header`) и ссылки на статьи, контактные данные администраторов сайта, ссылки на социальные сети (`footer`). Остальные шаблоны расширяют базовый, наполняя его контентом. Страница авторизации изображена на рисунке 1.

Основной моделью приложения является Компания. Пользователи могут создавать свои Группы Компаний и Офисы компаний. Список компаний показан на рисунке 2.

Для каждой компании можно создавать Вакансии, Новости, назначать События, публиковать результаты работы компании. Так же в приложении реализовано Social Share для публикации ссылок на страницы компании и объекты, связанные с компанией, в социальных сетях, что позволяет ускорить процесс рекрутинга сотрудников. Страница самой компании содержит всю необходимую информацию,

чтобы посетитель сайта мог ознакомиться с данными компании. Страница просмотра компании изображена на рисунке 3.

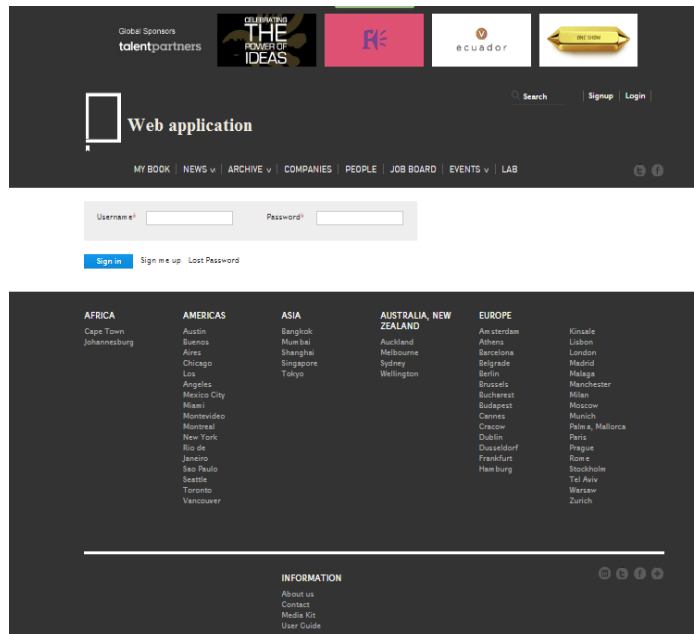


Рисунок 1 – Страница авторизации

## VELVET FILM PRODUCTION

Edit Remove  
Admins

## DROGA5

Edit Remove  
Admins

## MY COMPANIES

### BLABAL

Admins

Subscription status: **Not subscribed.**

[Subscribe](#)

Moderation status: **Rejected**

Profile

Followers **1**

Contacts **0**

Links **0**

Images **0**

News posted **0** + POST NEW ARTICLE

Work posted **0** + ADD WORK

Events posted **0** + LIST AN EVENT

Jobs posted **0** + POST A JOB

### LU

Admins

Subscription status: **Not subscribed.**

[Subscribe](#)

Moderation status: **Rejected**

Profile

Followers **1**

Contacts **0**

Links **0**

Images **0**

News posted **0** + POST NEW ARTICLE

Work posted **0** + ADD WORK

Events posted **0** + LIST AN EVENT

Jobs posted **0** + POST A JOB

Рисунок 2 – Список групп и офисов компаний

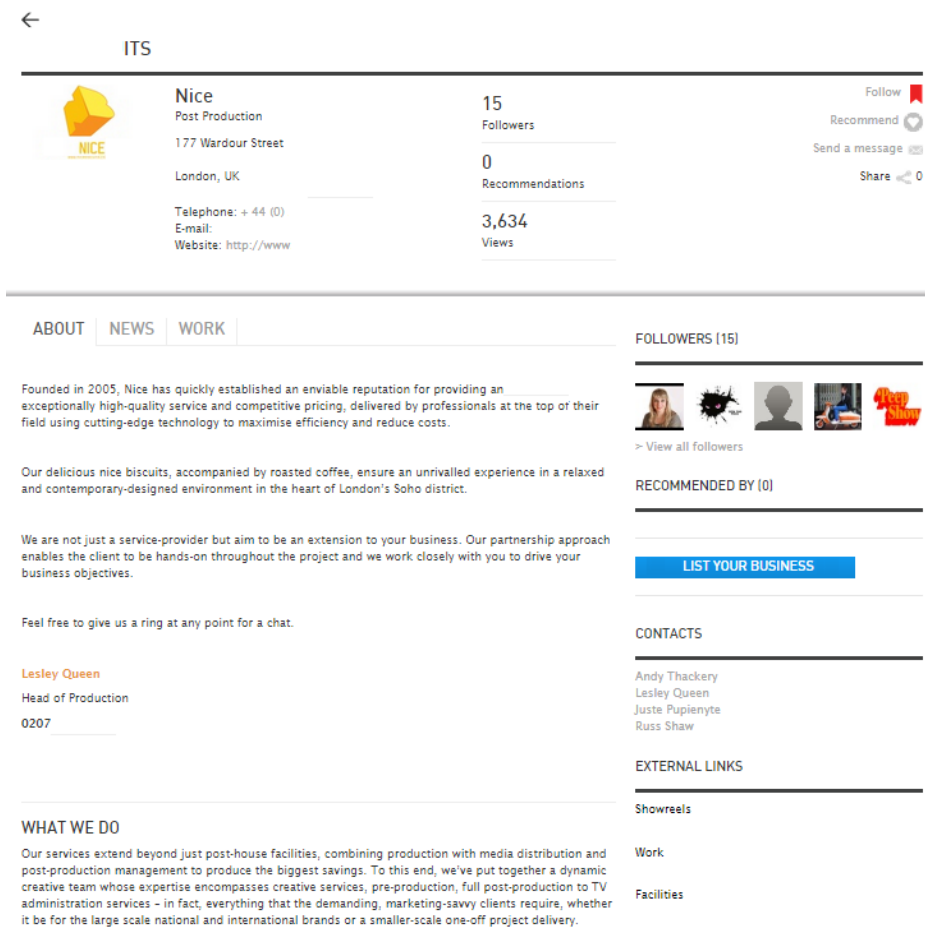


Рисунок 3 – Страница просмотра компании

**Н.А. Навныко (УО «ГГУ им Ф. Скорины», Гомель)**

Науч. рук. **В.Д. Левчук**, канд. техн. наук, доцент

## **АРХИТЕКТУРА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ РЕКРУТИНГА СОТРУДНИКОВ**

Разработанное приложение основано на CMS фреймворке Django. Выбор был сделан именно в сторону Django из-за ряда преимуществ. Во-первых, Django написан на языке Python – одном из наиболее распространённых интерпретируемых языков. Python позволяет выполнять большое количество разнообразных задач, поскольку является объектно-ориентированным языком программирования. Также он отличается динамичностью и кроссплатформенностью, то есть без проблем переносится на другую платформу (реализован практически на всех ОС и платформах). Во-вторых, Django предоставляет возможности использования ORM (Object-relationship mapper), что позволяет в большинстве случаев обходиться без SQL-синтаксиса в выражениях, что автоматически снижает риск появления SQL-injection уязвимости. Достоинствами

фреймворка являются автоматически генерируемая административная часть, множество классов для форм, представлений и моделей, позволяющие быстро и эффективно создать качественный сайт.

Django строит приложение на основе архитектуры MTV (Модель-Шаблон-Представление), которая соответствует шаблону MVC, где часть представлений представлена шаблонами, а часть контроллеров – представлениями django.

Шаблон Django – текстовый файл, содержащий html теги и `templatetags` Django. Шаблоны принято наследовать от базового. В нашем приложении базовый шаблон включает в себя основное меню сайта и баннеры (header) и ссылки на статьи, контактные данные администраторов сайта, ссылки на социальные сети (footer). Остальные шаблоны расширяют базовый, наполняя его контентом. Шаблон – это просто текстовый файл. Шаблон содержит переменные, которые будут заменены значениями при выполнении шаблона, и теги, которые управляют логикой шаблона.

Представления в Django – это контроллеры, обработчики запросов. Каждое представление закрепляется за одним или несколькими URL. В сам контроллер запрос поступает с дополненными Django атрибутами. Каждое представление должно возвращать http ответ (HTTPResponse), принимаемый браузером. Представления в Django можно описать функциями или классами, расширяющими базовые (ModelView, TemplateView и т. д.).

Веб-приложение подразделяется на две составные части: административную и основную.

Административная часть в Django Framework создается автоматически (если проект был создан командой `python manage.py startproject`). Административная часть приложения предназначена для управления данными. При регистрации модели администратору сайта доступен список объектов этой модели, которые он может редактировать или создавать новые объекты. Административная часть в Django Framework может быть изменена под любые нужды. В текущем приложении она используется для управления данными из базы данных, назначения прав пользователям.

Основная часть приложения (интерфейс) создается непосредственно шаблонами, представлениями и формами. Описывая формы Django, программист уменьшает количество кода HTML и получает возможность легко составлять структуру формы и методы ее обработки. Формы Django также позволяют автоматически осуществлять валидацию данных через описанные поля форм и их классы.

Детали реализации компонентов приложения обсуждаются в докладе.

**Д.А. Никульшин (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **В.А. Дробышевский**, ст. преподаватель

## **ПРЕИМУЩЕСТВА ИСПОЛЬЗОВАНИЯ ANGULARJS ПЕРЕД JAVASCRIPT**

JavaScript изначально позиционируется как простой язык, его синтаксис сильно приближён к языку Java, но несмотря на это, JavaScript является в большей степени хаотичным языком программирования. Причины этого лежат в архитектурных проблемах. Также важными факторами является то, что язык JavaScript изначально разрабатывался как функциональный, однако концепция была внезапно сменена, а также то, что времени на разработку языка было потрачено немного. В результате, в конечном языке присутствует местами запутанный синтаксис, работа некоторых операторов часто непредсказуема, есть большое число проблем с областью видимости переменных.

Несмотря на все недостатки языка JavaScript, была осуществлена масса попыток исправить эти недостатки, но больше всех на тот момент преуспел программист, разработавший библиотеку jQuery. jQuery решала не только проблему «одноразового» кода, но и учитывала вопросы кроссбраузерности. Данная библиотека также затрагивала синтаксис JavaScript, упрощая его и делая интуитивно понятным. jQuery позволила сократить большие конструкции и обеспечить стабильное выполнение кода. С каждым новым релизом в библиотеке jQuery появлялось больше функций, она совершенствовалась, а программисты получали больше возможностей. На данный момент библиотека jQuery является хорошим инструментом каждого web-программиста, а представить сайт, который не использовал бы данной библиотеки (либо её аналогов) трудно. Разработка jQuery дала толчок к развитию самого языка JavaScript. Так появились различные JavaScript-движки, такие как V8 от Google, который дал возможность создавать сложные и тяжёлые JS приложения. Подобные разработки стали подспорьем для создания принципиально новых технологий на базе JavaScript.

На смену библиотекам типа jQuery разрабатывались принципиально новые движки – фреймворки, реализующие паттерн MVC (MVVM, MVP). Фреймворки стали приносить альтернативные взгляды на устоявшихся в умах программистов вещи. Преимущества фреймворков можно заметить сразу. Одним из самых важных является избавление от рутинного кода, перемещающегося от проекта к проекту. Фреймворки предоставляют разработчикам каркас будущего приложения



и решение задач, часто встречающихся в большинстве проектов. К примеру, отпадает необходимость держать в уме алгоритм передачи данных от клиента на сервер, так как данная функция уже будет реализована авторами фреймворка. Вместо подобной рутинной работы разработчику предлагается сосредоточиться непосредственно функционалом своего приложения. Другим немаловажным плюсом фреймворков является стандартизация кодирования. Разработчики должны соблюдать стандарты и правила кодирования, если используют в проекте готовые каркасы, что даст возможность другим программистам быстрее и лучше понимать структуру этого приложения.

Проект Angular.JS впервые был представлен в 2009 г. Успех проекта был обеспечен компанией-разработчиком – Google. Angular.JS имеет низкий порог вхождения по сравнению со многими подобными решениями. Документация фреймворка хорошо структурирована, присутствуют примеры кода. Angular.JS естественным образом расширяет язык разметки HTML, вводя дополнительные директивы. Angular.JS позволяет очень просто решать любые рутинные задачи. К примеру, чтобы вывести содержимое какого-либо объекта, программисту не требуется писать громоздкие конструкции, а нужно заключить выражение в двойные фигурные скобки непосредственно в коде HTML. Например, результатом выполнения выражения  $1 + 2 = \{\{1 + 2\}\}$  будет: « $1 + 2 = 3$ ».

Одной из ключевых возможностей Angular.JS являются директивы. Директивы Angular.JS позволяют программисту описать поведение отдельных элементов и расширить синтаксис HTML. В Angular.JS входит лишь базовый набор директив, однако, нет никаких ограничений на добавление своих собственных, либо использования чужих директив, что делает выбор фреймворка Angular.JS логичным и правильным вместо использования нативного JavaScript.

Однозначно, Angular.JS является предельно простым, качественным и интересным решением, которое без сомнений стоит применять в своих проектах.

**В.С. Новиков (УО «ГГТУ им. П.О. Сухого», Гомель)**

Науч. рук. **Т.В. Тихоненко**, канд. физ.-мат. наук

## **LI-FI – НОВАЯ СИСТЕМА ПЕРЕДАЧИ ДАННЫХ**

В настоящее время в мире повсеместно используется такая технология беспроводной передачи информации, как Wi-Fi. Но она имеет

ряд недостатков, поэтому набирают популярность исследования о возможности использования света в передаче данных.

Технология Li-Fi, сокращенно от Light Fidelity, представляет собой оптическую технологию беспроводной передачи информации, представленную профессором Эдинбургского университета Харальдом Хаасом в 2011 году.

В отличие от сетей Wi-Fi, в которых данные передаются по радиоканалам, Li-Fi предполагает обмен информацией при помощи света. Для этого предлагается с высокой частотой изменять интенсивность свечения светодиодных ламп (LED): колебания происходят настолько быстро, что попросту незаметны для человеческого глаза. Для регистрации закодированных таким образом пакетов данных применяется специальный фотодетектор [1].

Передача данных при помощи света, к сожалению, пока еще используется для подключения конечного оборудования потребителей к Интернету крайне и крайне редко. Но, благодаря работе исследователей из Оксфордского университета, разрабатывается совершенно новый и более совершенный способ передачи данных. И это не будет обычным оптическим каналом на основе оптоволоконного кабеля, это будет реализовано при помощи новой Li-Fi-системы, излучающей свет в помещении, которая может обеспечить скорость обмена информацией более чем 100 гигабит в секунду.

Новая система внутренней оптической связи пока еще не может рассматриваться как полноценная замена беспроводной радиосвязи Wi-Fi. Однако, она в теории может обеспечить скорость передачи до 3 терабит в секунду и это позволяет уже рассматривать возможности применения оптических коммуникаций для расшивки узких мест в существующих сетях. Кроме этого, в случае использования оптических коммуникаций нет нужды беспокоиться относительно наличия свободного радиочастотного канала.

Система, как и другие Li-Fi-системы, использует нечто вроде базовой станции, устанавливаемой на потолке помещения, к которой подводится оптоволоконная линия. Эта станция излучает свет, принимаемый приемником, подключенным к компьютеру, и принимает передаваемые им же данные, осуществляя обмен информацией, на основе которого стоит весь Интернет. В отличие от обычных Li-Fi-систем, которые используют свет видимого диапазона, излучаемый светодиодными осветительными приборами, новая система работает при помощи инфракрасного света с длиной волны в 1550 нанометров, который широко используется в области оптических коммуникаций.

Главной уловкой в новой технологии является использование так называемого голографического луча, при помощи которого удастся

направить весь свет в строго заданном одном или нескольких направлениях. Для получения такого луча используется специальная жидкокристаллическая матрица, подобная матрицам, используемым в некоторых типах проекторов, которая при помощи явления дифракции отражает свет в заданном направлении [2].

В этой технологии немаловажное значение имеет ширина области охвата приемно-передающих устройств, которые могут обеспечить качественную реализацию технологии мультиплексирования при помощи разделения длин волн света. При 60-градусной области охвата приемно-передающих устройств исследователям удалось реализовать передачу данных по шести независимым потокам, скорость каждого из которых составила 37,4 гигабита в секунду, следовательно, скорость совокупного канала составила 224 гигабита в секунду. При значении угла области охвата в 36 градусов удастся реализовать только три независимых потока, а общая скорость канала в этом случае равна 112 гигабитам в секунду.

Для работы системы пока еще требуется, чтобы приемник и передатчик находились в зоне прямой видимости и находились в неподвижном положении. Но оксфордские исследователи уже разрабатывают следящую систему, которая позволяет отслеживать перемещение компьютера, ноутбука к примеру, и перестраивать жидкокристаллические матрицы приемника и передатчика так, чтобы обеспечить надежную и высокоскоростную связь [3].

Приемные датчики, подключенные к компьютерам или другим цифровым электронным устройствам, позволяют получать информацию тогда, когда на них падает прямой свет от источника Li-Fi. Это, конечно накладывает некоторые ограничения на возможности использования данной технологии, но в конечном счете, такая оптическая связь является несколько безопасней, нежели традиционный Wi-Fi, сигналы которого можно перехватить из любой точки, находящейся в радиусе действия оборудования. Для того, чтобы перехватить информацию, передаваемую через Li-Fi, злоумышленнику потребуется разместить свое шпионское оборудование непосредственно около источника света, что может привести к серьезным для него трудностям.

Помимо вышесказанного, технология оптической беспроводной связи Li-Fi может без ограничений использоваться в местах, где запрещено использование оборудования, излучающего посторонние радиоволны, которые могут нарушить нормальную работу критичного оборудования. К таким местам, безусловно, относятся реанимационные палаты медицинских учреждений, салоны самолетов и некоторые другие места [2].

В конечном итоге можно сказать, что Li-Fi система из-за своих недостатков не будет иметь такую-же популярность как Wi-Fi, но будет использоваться в некоторых сферах, например в медицине и авиации, в которых использование радиоволн не рекомендуется.

### Литература

1. Новостной Интернет-портал Computerra [Электронный ресурс] / ООО «Компьютерра-Онлайн». – Москва, 1997. – Режим доступа : [www.computerra.ru](http://www.computerra.ru) – Дата доступа : 14.03.2015.

2. DailyTechInfo - Новости науки и технологий, новинки техники. [Электронный ресурс] / Москва, 2009. – Режим доступа : [www.dailytechinfo.org](http://www.dailytechinfo.org) – Дата доступа : 14.03.2015.

3. Новостной Интернет-портал InternetNovosti.com [Электронный ресурс] / Анапа, 2004. – Режим доступа : [www.internetnovosti.com](http://www.internetnovosti.com) – Дата доступа : 14.03.2015.

**В.В. Осипенко (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **П.Л. Чечет**, канд. техн. наук, доцент

### ОБЗОР ВОЗМОЖНОСТЕЙ DIRECTX 12

20 марта, в рамках конференции GDC 2014, состоялся анонс следующей версии DirectX 12. Основным ядром DirectX является Direct3D, именно на нем создаются наиболее критичные компоненты игровых приложений. Команда разработки внесла ряд изменений в Direct3D, в результате которых возросла скорость и эффективность многих графических операций. Эти изменения позволяют создавать более детальные сцены и достичь полного использования возможностей современных GPU. Но эти возможности появятся не только в «хай-энд» игровых компьютерах. Direct3D 12 будет работать на всех устройствах Microsoft. Телефоны, планшеты, ноутбуки, десктопы и конечно, Xbox One, все они позволят использовать API Direct3D 12. В первую, и самую главную очередь, это более низкий уровень абстрагирования оборудования. Это позволяет играм значительно улучшить поточную масштабируемость и уровень использования GPU. В дополнение, игры получают выигрыш от таблиц дескрипторов и объектов состояния конвейера. Это конечно не всё, Direct3D включает набор новых возможностей для конвейера рендера, которые могут в разы увеличить эффективность расчетов прозрачности, определения коллизий и геометрической выбраковки. Само собой разумеется, что

API хорошо только тогда, когда есть инструменты, которые помогают его эффективно использовать. DirectX 12 будет содержать ряд отличных инструментов для Direct3D сразу после выхода DirectX 12 в свет.

Положительный момент заключается в том, что DirectX 12 будет работать на многих существующих видеокартах. Скриншоты на рисунке 1 были сделаны на реальных Direct3D 12 приложениях с реальной реализацией рантайма и драйверов Direct3D 12 для популярного приложения для тестирования производительности видеокарт и процессора 3D Mark. Для любителей игр приятной новостью будет тот факт, что, несмотря на то, что Direct3D 11 позволяет использовать поточное масштабирование, но из-за некоторых накладок, связанных с рантаймом и драйверами, существует холостое время ожидания потоков и ядер. После переноса приложений на Direct3D 12 планируются значительные улучшения – до 50 % прироста к утилизации CPU и улучшение распределения работы между потоками.

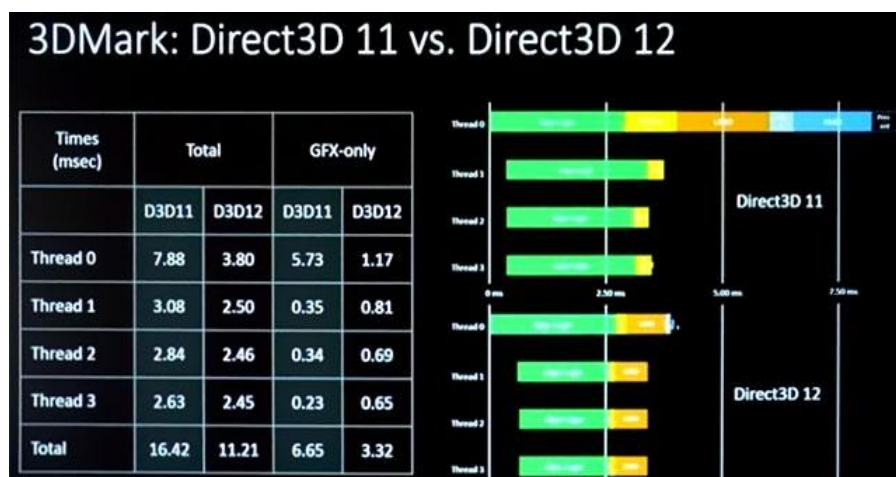


Рисунок 1 – Показатели тестов Direct3D 11 и Direct3D 12

Direct3D 12 отходит от старой модели программирования Direct3D 11, позволяя приложению быть значительно ближе к железу как никогда ранее. Мы пересмотрели значительное количество областей API, ключевыми из которых являются механизмы представления состояний конвейера, методы отправки заданий и доступ к ресурсам.

Direct3D 11 позволяет манипулировать состояниями конвейера через большой набор перекрестных объектов. Например, состояние входного ассемблера, состояние пиксельного шейдера, состояние растеризатора и состояние выходного смесителя, все они могут быть изменены независимо. Этот механизм предоставляет удобный, относительно высокоуровневый уровень представления графического конвейера, но, к сожалению, он не очень хорошо ложится на современное оборудование.

В первую очередь из-за того, что существуют взаимозависимости между многими состояниями. Например, многие GPU комбинируют состояние пиксельного шейдера и выходного смесителя в одно аппаратное представление. Direct3D 11 API позволяет менять их состояние независимо, но драйвер не может разрешить это пока не будет знать что состояние завешено, а это не возможно до начала отрисовки. Это приводит к задержкам при установке состояний аппаратной части устройства, означает издержки и меньшее количество вызовов отрисовки на каждый кадр. Direct3D 12 позволяет решить эту проблему унифицируя большую часть состояния конвейера в неизменяемый объект состояния конвейера (ОСК), который фиксируется сразу же после его создания. Это позволяет железу и драйверу немедленно превратить ОСК в соответствующие аппаратные инструкции, а состояние необходимо для выполнения работы GPU. При этом можно динамически выбирать, какой ОСК на текущий момент необходимо использовать. Теперь остается только скопировать в аппаратные регистры небольшой объем заранее просчитанного состояния, вместо вычисления аппаратного состояния на лету. Это означает значительно меньший объем накладных расходов между вызовами отрисовки, и большее количество вызовов отрисовки на каждый кадр.

В Direct3D 11 все задания отправляются с помощью непосредственного контекста, который представляет один ряд команд, отправляющихся в GPU. Для многопоточного масштабирования производительности игры используют отложенные контексты, но, как и в случае ОСК, отложенные контексты не соответствуют реальному положению вещей в аппаратном устройстве.

Direct3D 12 предоставляет новую модель для отправки заданий, базирующуюся на списках команд. Они содержат всю необходимую информацию для исполнения в GPU. В список команд входит ОСК, текстурные и буферные ресурсы, а также аргументы для команды отрисовки. В связи с тем, что список команд автономен и не содержит состояния, драйвер может заранее просчитать необходимые GPU команды в независимом потоке. Необходим лишь процесс сериализации при финальной отправке списка команд в GPU, который производится через очередь команд, и этот процесс очень эффективен.

В дополнение к спискам команд, в Direct3D 12 включен механизм вторичного предварительного вычисления с помощью пакетов. В отличие от списков команд, которые полностью автономны, и обычно создаются, отправляются для выполнения, а затем уничтожаются, пакеты предоставляют некую форму независимости от состояния и подразумевают повторное использование. Например, если игре необходимо

отобразить две модели персонажей с разными текстурами, одним из подходов будет запись списка команд с двумя наборами идентичных вызовов отрисовки. Другим подходом является запись одного пакета, который отрисует одного персонажа, а затем «перемотки назад» пакета дважды в списке команд, используя разные ресурсы. В последнем случае драйверу необходимо вычислить инструкции только единожды, а создание списка команд по существу равно двум вызовам функций с низкой стоимостью.

Привязка ресурсов в Direct3D 11 хорошо абстрагирована и достаточно удобна, но при этом многие возможности современного железа остаются не у дел. В Direct3D 11 игра создает объекты «представления», затем привязывает эти представления к «слотам» на различных этапах шейдера в конвейере. Шейдеры в свою очередь читают данные из этих явно привязанных слотов, которые фиксированы в момент отрисовки. Такая модель означает, что в случае, если игре надо сделать отрисовку, используя другой набор ресурсов, необходимо сделать повторную привязку представлений к другим слотам и заново вызвать отрисовку. Это еще один пример накладных расходов, которые могут быть устранены если полностью использовать современные возможности аппаратного оборудования.

Direct3D 12 меняет модель привязки, чтобы соответствовать современным возможностям и значительно увеличивает производительность. Вместо того чтобы требовать автономные ресурсные представления и явную привязку к слотам, Direct3D 12 предоставляет кучу дескрипторов, где игра создает свои представления ресурсов. Такой механизм позволяет GPU напрямую и авансом записывать аппаратное представление описания ресурса в память. Для декларации того, какой ресурс будет использован в конвейере для конкретного вызова отрисовки, игра указывает одну или несколько дескрипторных таблиц, которые представляют суб-диапазон полной кучи дескрипторов. Так как эта куча заранее загружена необходимыми специфичными для оборудования данными, изменение таблицы дескрипторов является очень дешевой операцией.

В дополнение к улучшениям, связанным с дескрипторной кучей и таблицами, Direct3D 12 позволяет динамически индексировать ресурсы в шейдерах. Это предоставляет беспрецедентную гибкость и открывает дорогу к новым техникам рендера. Например, современные движки рендера, основанные на отложенном подходе, часто кодируют материал или идентификатор объекта какого-то типа в предварительный g-буфер. В Direct3D11 такие движки должны быть осторожны в использовании слишком большого количества материалов, так как включение множества g-буферов может значительно снизить скорость

финального прохода отрисовки. Вместе с динамически финализируется так же быстро, как и с десятью.

**В.В. Осипенко (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **П.Л. Чечет**, канд. техн. наук, доцент

## **НОВАЯ ВЕРСИЯ DIRECTX**

В прошлом году состоялся анонс следующей версии популярного компонента DirectX: DirectX 12. Сердцем DirectX 12 является новая, 12-я версия компонента Direct3D – самой, пожалуй, главной части DirectX, благодаря которой мы и можем наблюдать всю красоту в играх. Программисты, трудившиеся над Direct3D 12, произвели существенную переработку этого компонента, что вылилось в эффективность и скорость его работы. Direct3D 12 позволит на полную мощь и максимально эффективно использовать возможности современных графических процессоров, что, в свою очередь, позволит разработчикам создавать более насыщенные сцены с множеством объектов. Корпорация Microsoft сообщила, что Direct3D 12 рассчитан на работу на устройствах разного «калибра», начиная со смартфонов и планшетов, заканчивая ноутбуками и стационарными компьютерами и консолью Xbox One.

Объясняет Microsoft всё вполне логично. Во-первых, Direct3D 12 предоставляет возможность более низкоуровневого доступа к аппаратному обеспечению, чем раньше, что позволит играм значительно улучшить масштабируемость при использовании многоядерных процессоров и снизить на них общую нагрузку. Во-вторых, по заверениям Microsoft, благодаря дескрипторным таблицам и новым высокотехнологичным методам рендеринга также снизится нагрузка и на графический процессор.

После переноса приложений на Direct3D 12 планируются значительные улучшения – до 50 % прироста к использованию CPU и улучшение распределения работы между потоками. Причинами подобных улучшений стал отход от старой модели программирования Direct3D 11, позволяя приложению быть значительно эффективнее использовать возможности современного аппаратного обеспечения, которые значительно возросли за последние годы.

Microsoft заявляет, что приобретать новые компьютеры или видеокарты необходимости нет, DirectX 12 будет поддерживаться примерно 80-ю процентами продаваемых сегодня игровых компьютеров. Microsoft свяжется со своими партнёрами на предмет внедрения информации



о поддержке DirectX 12 их продуктами. Первые игры с поддержкой DirectX 12 должны появиться к праздничному сезону 2015 г., а некоторые разработчики движков, в частности, Epic Games, уже заявили, что внедрят в них поддержку DirectX 12.

**Р.С. Панасюк (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Д. Левчук**, канд. техн. наук, доцент

## **АВТОМАТИЗАЦИЯ СИСТЕМЫ ОТЧЕТНОСТИ ПО ИНТЕРНЕТ РЕКЛАМЕ**

В современном мире маркетинг старается использовать как можно больше каналов для привлечения новых клиентов, самым динамичным из которых остается интернет-реклама, которую можно встретить на практически любом интернет-ресурсе.

Для автоматизации данной предметной области было разработано веб-приложение, являющееся DMP-системой. Это многофункциональная система, которая позволяет рекламодателям, агентствам, издателям и остальным сторонам хранить и систематизировать имеющиеся у них данные первого и второго порядка и дополнять их данными третьего порядка.

DMP позволяет оптимизировать медиазакупки и планирование рекламных кампаний за счет поведенческих моделей или расширения аудитории через look-alike-моделирование (поиск пользователей, похожих на тех, которые уже стали клиентами). Рекламодатели и агентства обычно используют DMP для того, чтобы более эффективно покупать, в то время как для издателей DMP – это, в первую очередь, сегментация своей аудитории и более эффективная продажа своих данных.

Например, у маркетолога ретейл-компании есть база лояльных клиентов некоторого предприятия, которым были выданы персональные карточки постоянных клиентов. На текущий момент предприятие не может с ними взаимодействовать в интернете (исключение: email-рассылка, или использование соцсетей). Основная информация, которой обладает предприятие, это база email-адресов, а также данные о покупках в CRM. Задача предприятия — выйти за пределы рассылки и соцсетей для взаимодействия со своими лояльными клиентами в интернете.

Здесь на помощь приходит DMP. Предприятие внедряет изображение для отслеживания в каждое письмо (как в ссылку, так и в картинку, в которой содержится номер карты), и при открытии письма в браузере или при переходе по ссылке синхронизируются cookie пользователя в базе DMP с ID базы и номером карты.

Но письма читают далеко не все. Что делать, чтобы синхронизировать больше людей? Здесь следует использовать сервисы или акции, в которые необходимо вводить номер карты для активации аккаунта или регистрации. Сервисами пользуются большинство постоянных пользователей, а в акциях обычно участвуют большое количество человек. То есть всё сводится к вопросу реализации: как организовать возможность сбора номеров карточек через интернет.

Если рассматривать базу постоянных клиентов в сотни тысяч, то успехом будет являться показатель в 30 % синхронизированных пользователей. Если бизнес меньше, то процент конверсии необходимо увеличивать. Чем меньше аудитория, тем сложнее ее идентифицировать в интернете.

После того, как было синхронизировано достаточное количество пользователей – представителей аудитории предприятия, они могут проводить персонализированные рекламные кампании через протокол RTB (Real Time Bidding) на всех просторах интернета. Зная о том, какие потребительские предпочтения у аудитории, предприятие может делать обращения, которые будут вызывать повышенный отклик на рекламу.

Важным компонентом в такой сложной системе, как DMP, является интерфейс взаимодействия с конечным пользователем. Он должен предоставлять простые и понятные методы доступа к данным маркетинговых кампаний, а так же уметь строить отчеты, визуализировать данные и проводить статистический анализ для получения более точных и наглядных результатов.

В разработанной системе отчетности для DMP используются современные подходы к разработке веб-приложений. Ключевым из использованных подходов является одностраничность – подход, при котором каждая страница веб-приложения генерируется на клиенте, а сервер, главным образом, выступает в роли API к базе данных, содержит разного рода валидации и некоторую часть бизнес – логики. Большая часть всей логики приложения хранится на стороне клиента.

Так же в архитектуру системы был положен принцип REST (REpresentational State Transfer), который описывает архитектуру построения веб-сервиса, в котором сервер не хранит информацию о состоянии клиента; вносится понятие ресурса, определяющего некоторую модель данных и операций над ними; используются описанные в протоколе HTTP методы для совершения операций над ресурсами вместо различных постфиксов или url-параметров.

Таким образом реализуется масштабируемая, надежная и простая система, которую легко кешировать и тестировать. Но есть и негативные стороны, например, требования отказа от хранения состояния

идут вразрез с моделью построения большинства моделей веб-аутентификации. В данном проекте был достигнут компромисс между REST и классическими технологиями: сервер хранит состояние сессии и пользователя для обеспечения аутентификации. Для этого в каждый запрос добавляется http-заголовок x-sessionid однозначно идентифицирующий пользователя.

Для поддержки актуального состояния схемы базы данных на всех окружениях был использован механизм миграций, предоставленный компонентом Django South. С помощью этой технологии удалось полностью автоматизировать процесс развертывания базы данных и упростить поддержку состояния схемы, полностью исключив вмешательство программиста.

Таким образом была спроектирована современная масштабируемая автоматизированная система отображения отчетов по интернет-рекламе для DMP-системы.

**Р.С. Панасюк (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Д. Левчук**, канд. техн. наук, доцент

## **ИСПОЛЬЗОВАНИЕ ПРИНЦИПА REST ДЛЯ ОРГАНИЗАЦИИ ОБМЕНА ДАННЫМИ КЛИЕНТА И СЕРВЕРА В ПРИЛОЖЕНИИ АВТОМАТИЗАЦИИ ОТЧЕТНОСТИ ИНТЕРНЕТ-РЕКЛАМЫ**

При разработке системы отчетности интернет-рекламы для обеспечения удобного клиент-серверного интерфейса обмена данными был выбран подход REST, который, в теории, обеспечивает лучшие показатели масштабируемости и надежности, чем использование различных бессистемных подходов, широко применяемых в классических многостраничных динамических веб-приложениях.

REST (сокр. от англ. Representational State Transfer – «передача состояния представления») – метод взаимодействия компонентов распределённого приложения в сети Интернет, при котором вызов удаленной процедуры представляет собой обычный HTTP-запрос (GET, POST, PUT, DELETE; такой запрос называют REST-запрос), а необходимые данные передаются в качестве параметров запроса. Этот способ является альтернативой более сложным методам, таким как SOAP, CORBA и RPC.

В широком смысле REST означает концепцию построения распределённого приложения, при которой компоненты взаимодействуют наподобие взаимодействия клиентов и серверов во Всемирной паутине.

Хотя данная концепция лежит в самой основе Всемирной паутины, термин REST был введён Роем Филдингом лишь в 2000 г. Филдинг

описал концепцию построения распределённого приложения, при которой каждый запрос (REST-запрос) клиента к серверу содержит в себе исчерпывающую информацию о желаемом ответе сервера (желаемом репрезентативном состоянии), и сервер не обязан сохранять информацию о состоянии клиента («клиентской сессии»).

В качестве необходимых условий для построения распределённых REST-приложений Филдинг перечислил следующие: клиент-серверная архитектура; сервер не обязан сохранять информацию о состоянии клиента; в каждом запросе клиента должно явно содержаться указание о возможности кэширования ответа и получения ответа из существующего кэша; клиент может взаимодействовать не напрямую с сервером, а с произвольным количеством промежуточных узлов. При этом клиент может не знать о существовании промежуточных узлов, за исключением случаев передачи конфиденциальной информации. Филдинг приводил URI в качестве примера формата запросов к серверу, а в качестве примера ответа сервера форматы HTML, XML и JSON, различаемые с использованием идентификаторов MIME.

В разработанном приложении были использованы лишь некоторые подходы REST по той причине, что на практике требования накладываемые данной концепцией зачастую противоречат требованиям инфраструктуры. В частности архитектурой не предусматривается реализации аутентификации, точнее она не может быть реализована обычным способом, т. к. это нарушит требование о хранении состояния клиента на сервере.

В реальных приложениях многими требованиями можно пренебречь, сфокусировавшись на более важных с архитектурной точки зрения. В данном веб-приложении был использован подход REST наряду с использованием веб-сессий, что оказалось удачным архитектурным решением, сильно упростившим поддержку системы.

**В.М. Пархамович (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Леванцов**, ст. преподаватель

## **ИСПОЛЬЗОВАНИЕ АРМ ДЛЯ АВТОМАТИЗАЦИИ УЧЕТА УСЛУГ НА ПРЕДПРИЯТИИ**

В настоящее время на рынке существует большое количество решений для автоматизации практически любой деятельности организаций.

Говорить о том, что каждое из этих решений идеально впишется в рамки бизнес процессов, протекающих в любой организации, не

приходится. Стандартные решения автоматизации способны удовлетворить многие, но не все требования организаций. В независимости от отрасли, или сферы ведения хозяйственной деятельности одно и то же готовое средство может оказаться излишне функциональным – в лучшем случае, или иметь изъян в виду своей ограниченности.

Сейчас в мире идет процесс максимального внедрения компьютерных систем и программ во все сферы производства, создание прикладного программного обеспечения. Сегодня уже не найти сферу человеческой деятельности, где бы не использовались информационные технологии. Но это не значит, что компьютеры могут полностью заменить человека на сегодняшний день – это не реально. Для работы в разных сферах деятельности необходимо участие человека, умение проводить аналитические расчеты, знание логических взаимосвязей, заполнение форм бухгалтерской отчетности и многого другого.

Сейчас на очень многих предприятиях реализуется концепция распределенных систем управления народным хозяйством. В них предусматривается локальная, достаточно полная и в значительной мере законченная обработка информации на различных уровнях иерархии. В этих системах организуется передача снизу вверх только той части информации, в которой имеется потребность на верхних уровнях. При этом значительная часть результатов обработки информации и исходные данные должны храниться в локальных банках данных.

Для реализации идеи распределенного управления потребовалось создание для каждого уровня управления и каждой предметной области автоматизированных рабочих мест на базе профессиональных персональных компьютеров.

Автоматизированное рабочее место (АРМ), или, в зарубежной терминологии, «рабочая станция» (work-station), представляет собой место пользователя-специалиста той или иной профессии, оборудованное средствами, необходимыми для автоматизации выполнения им определенных функций. Такими средствами, как правило, является ПК, дополняемый по мере необходимости другими вспомогательными электронными устройствами, а именно: дисковыми накопителями, печатающими устройствами, оптическими читающими устройствами или считывателями штрихового кода, устройствами графики, средствами сопряжения с другими АРМ и с локальными вычислительными сетями и т. д.

Например, в сфере экономики на таких АРМ можно осуществлять планирование, моделирование, оптимизацию процессов, принятие решений в различных информационных системах и для различных сочетаний задач. Для каждого объекта управления необходимо предусматривать

АРМ, соответствующие их значению. Однако принципы создания любых АРМ должны быть общими:

- системность;
- гибкость;
- устойчивость;
- эффективность.

На производственных предприятиях АРМ являются важной структурной составляющей АСУ как персональное средство планирования, управления, обработки данных и принятия решений.

Проектирование является основополагающей стадией в разработке программного обеспечения. Поэтому очень важно на этом этапе учесть все требования к программному продукту, создать соответствующую предметной области базу данных, предусмотреть связь между данными, обеспечить их непротиворечивость и целостность. Выбор средств реализации проекта также является ответственным шагом в проектировании прикладных приложений, так как используемые средства разработки должны обеспечивать возможность создания приложений с учетом всех заявленных требований от заказчика, быть современными, и находиться в ценовой категории тех, которые может позволить себе заказчик.

Основная задача программного обеспечения освободить руководителя фирмы, который выполняет функции специалиста автоматизации учета услуг от рутинной работы по просмотру и выборке нужных записей, автоматизировав этот процесс, а также предоставить возможность быстрого составления любых отчетов, сформированных по требуемому запросу.

Созданные формы, запросы и отчеты позволяют быстро и эффективно обновлять данные, получать ответы на вопросы, осуществлять поиск нужных данных, анализировать данные, печатать отчеты, карточки учета и другие нужные формы. Использование в работе различных конструкторов и мастеров создания форм позволило быстро создавать необходимые выходные формы, не затрачивая на это много времени и сил.

Приложение «Учет услуг» предназначено для автоматизации деятельности учета услуг организации. Система позволит обеспечить многопользовательский доступ к единой базе данных организации, при этом будет возможна одновременная работа нескольких пользователей с информацией, что позволит гибко распределить обязанности между сотрудниками (например, регистрация новых сотрудников, модификация информации, поиск, формирование отчетов и прочее). При работе в многопользовательском режиме пользователям могут назначаться права доступа к информации, например, только руководитель

может редактировать записи. Однако система не исключает возможность работы и в локальном варианте – на одном рабочем месте.

Программа «Учет услуг» предназначена для облегчения условий труда сотрудников автоцентра. Наиболее рутинными и в то же время наиболее ответственными процессами являются:

- ввод метрики клиента в его учетную карточку;
- ведение архива учета услуг;
- подготовка различных отчетов данным клиентов (марка автомобиля, ремонтные работы, информация о клиенте).

Программный комплекс по автоматизации учета услуг полностью отвечает поставленным перед ним требованиям. Имеет дружественный интерфейс и не требует от пользователей специальной подготовки, не связанной с их профессиональными обязанностями.

**В.М. Пархамович (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Леванцов**, ст. преподаватель

## **АВТОМАТИЗАЦИЯ УЧЕТА УСЛУГ НА ПРЕДПРИЯТИИ**

Сейчас в мире идет процесс максимального внедрения компьютерных систем и программ во все сферы производства, создание прикладного программного обеспечения. Сегодня уже не найти сферу человеческой деятельности, где бы не использовались информационные технологии. Но это не значит, что компьютеры могут полностью заменить человека на сегодняшний день – это не реально. Для работы в разных сферах деятельности необходимо участие человека, умение проводить аналитические расчеты, знание логических взаимосвязей, заполнение форм бухгалтерской отчетности и многого другого.

Сейчас на очень многих предприятиях реализуется концепция распределенных систем управления народным хозяйством. В них предусматривается локальная, достаточно полная и в значительной мере законченная обработка информации на различных уровнях иерархии. В этих системах организуется передача снизу вверх только той части информации, в которой имеется потребность на верхних уровнях. При этом значительная часть результатов обработки информации и исходные данные должны храниться в локальных банках данных.

Для реализации идеи распределенного управления потребовалось создание для каждого уровня управления и каждой предметной области автоматизированных рабочих мест на базе профессиональных персональных компьютеров.

Автоматизированное рабочее место (АРМ) представляет собой место пользователя-специалиста той или иной профессии, оборудованное средствами, необходимыми для автоматизации выполнения им определенных функций. Для каждого объекта управления необходимо предусматривать АРМ, соответствующие их значению. Однако принципы создания любых АРМ должны быть общими: системность, гибкость, устойчивость, эффективность.

Приложение «Учет услуг» предназначено для автоматизации деятельности учета услуг организации. Система позволит обеспечить многопользовательский доступ к единой базе данных организации, при этом будет возможна одновременная работа нескольких пользователей с информацией, что позволит гибко распределить обязанности между сотрудниками (например, регистрация новых сотрудников, модификация информации, поиск, формирование отчетов и прочее). При работе в многопользовательском режиме пользователям могут назначаться права доступа к информации, например, только руководитель может редактировать записи. Однако система не исключает возможность работы и в локальном варианте – на одном рабочем месте.

**В.Э. Петрова (УО БрГТУ, Брест)**

Науч. рук. **В.И. Хвещук**, канд. техн. наук, доцент

## **СРЕДСТВА АВТОМАТИЗАЦИИ ИНФОРМАЦИОННОГО МОДЕЛИРОВАНИЯ ПРЕДПРИЯТИЙ**

Постоянное усложнение производственно-технических и организационно-экономических систем – фирм, предприятий, производств и других субъектов производственно-хозяйственной деятельности – и необходимость их анализа с целью совершенствования функционирования и повышения эффективности обуславливают необходимость применения специальных средств описания и анализа таких систем.

Создание современных автоматизированных систем обработки документированной информации (АСОДИ) представляет собой сложную задачу, решение которой требует применения специальных методик и инструментов. В последнее время среди системных аналитиков и разработчиков значительно вырос интерес к CASE–технологиям и инструментам, позволяющим максимально систематизировать и автоматизировать этапы разработки этих систем.

В рамках данной работы предложен вариант системы автоматизации проектирования (САП) в виде набора средств, ориентированных



на построение информационной модели предприятия как совокупности организационной структуры, схемы документооборота, задач обработки данных, которые предназначены для использования в процессе разработки концепции АСОДИ.

САП реализуется на основе клиент – серверной архитектуры и состоит из серверной и клиентской части. Серверная часть представлена базой данных реализованной с использованием Microsoft SQL Server, а клиентская часть создана на основе С#.

Клиентское приложение обеспечивает автоматизацию описания следующих компонентов предприятия: организационную структуру в виде иерархии подразделений, отдельное подразделение может состоять из групп сотрудников; документооборот в виде документов и взаимосвязей между ними, а также моделей документов; задач обработки данных и их взаимосвязей с документами; закрепление задач за сотрудниками предприятия; справочные и другие функции.

Предложенная САП разрабатывается в рамках дипломного проектирования, и на данный момент находится на стадии испытания и планируется внедрение.

**М.Н. Пилипейко (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **Н.А. Аксёнова**, ассистент

## **ПРОЕКТ ИССЛЕДОВАНИЯ МЕТОДА ОБРАТНОЙ ТРАССИРОВКИ ЛУЧЕЙ**

Задачей проекта является построение реалистичных изображений, используя метод обратной трассировки лучей.

Трассировка лучей (ray tracing) – метод машинной графики, позволяющий создавать фотореалистические изображения любых трехмерных сцен. Трассировка лучей моделирует прохождение лучей света через изображаемую сцену. Фотореализм достигается путем математического моделирования оптических свойств света и его взаимодействия с объектами. Сначала отдельные объекты располагаются в трехмерном пространстве-сцене, а также задаются физические и оптические свойства их поверхностей и их цвет. Затем определяется, где будут расположены источники света и наблюдатель. И наконец, с помощью программы трассировки лучей создается математическая модель сцены, света и наблюдателя, с помощью которой вычисляется цвет каждого пикселя графического изображения, получаемого на экране дисплея.

При практической реализации метода обратной трассировки вводятся ограничения. Некоторые из них необходимы, чтобы можно было в принципе решить задачу синтеза изображения, а некоторые ограничения позволяют значительно повысить быстродействие трассировки.

Ограничения при реализации трассировки.

Среди всех типов объектов выделяются такие, которые называем источниками света. Источники света могут только излучать свет, но не могут его отражать или преломлять.

Свойства отражающих поверхностей описываются суммой двух составляющих – диффузной и зеркальной.

В свою очередь, зеркальность также описывается двумя составляющими. Первая (reflection) учитывает отражение от других объектов, не являющихся источниками света. Строится только один зеркально отраженный луч  $r$  для дальнейшей трассировки. Вторая компонента (specular) означает световые блики от источников света. Для этого направляются лучи на все источники света и определяются углы, образуемые этими лучами с зеркально отраженным лучом обратной трассировки ( $r$ ). При зеркальном отражении цвет точки поверхности определяется собственным цветом того, что отражается.

При диффузном отражении учитываются только лучи от источников света. Лучи от зеркально отражающих поверхностей игнорируются. Если луч, направленный на данный источник света, закрывается другим объектом, значит, данная точка объекта находится в тени. При диффузном отражении цвет освещенной точки поверхности определяется собственным цветом поверхности и цветом источников света.

Для завершения трассировки вводится ограничение количества итераций (глубины рекурсии).

Расчет освещения сцены проходит по следующему алгоритму:

1. Отслеживаются лучи, проходящие из глаза наблюдателя через каждый пиксел экрана в сцену.

2. На каждой поверхности сцены, на которую попадает луч, формируются отраженный и преломленный лучи.

3. Каждый такой луч рекурсивно отслеживается, чтобы определить пересекаемые поверхности.

4. Ветвление прекращается если:

- луч вышел за пределы сцены;
- луч пришел на источник света;
- луч попал на непрозрачный диффузный рассеиватель;
- исчерпана память.

5. В результате для каждого пиксела строится дерево пересечений. Для каждого пиксела каждый луч должен быть проверяться на

пересечение с каждой поверхностью. Ветви такого дерева – распространение луча, а узлы – пересечения с поверхностями.

6. Закраска пиксела определяется прохождением по дереву и вычислением вклада каждой пересеченной поверхности в соответствии с используемыми моделями отражения.

Принцип работы обратной трассировки лучей изображен на рисунке 1, где  $S_1$  – блестящая прозрачная поверхность,  $S_2$  и  $S_4$  – рассеивающие непрозрачные поверхности,  $S_3$  – блестящая непрозрачная поверхность.

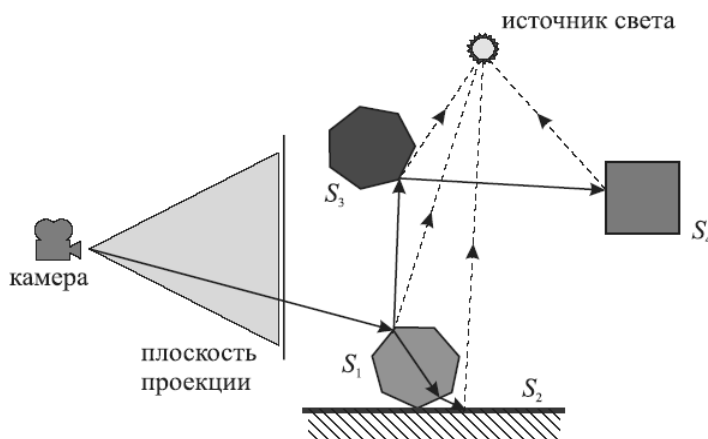


Рисунок 1 – Обратная трассировка лучей

Тени в алгоритме трассировки лучей учитываются следующим образом. Из каждой точки пересечения луча трассировки с поверхностью строятся дополнительные лучи по направлению к каждому источнику света. Если такой луч пересекает на своем пути какую-нибудь поверхность, то на точку, из которой был выпущен луч, падает тень от этой поверхности. Таким образом, все эти лучи вместе с данными о физических характеристиках объектов модели (цвет, прозрачность, зеркальность и т. д.) позволяют определить цвет и его интенсивность для каждой точки изображения. Изображение с построенными тенями выглядит гораздо реалистичнее, и, кроме того, тени очень важны для моделирования.

Объекты, освещенные одним источником света, выглядят не естественно, потому что в реальности объект освещается светом, отраженным от окружающих его объектов. Поэтому ввели понятие как рассеянный свет – это свет, равномерно и одинаково освещающий все поверхности объекта.

Поверхности могут направленно и диффузно пропускать свет. Направленное пропускание света происходит сквозь прозрачные вещества (стекло). Через них хорошо видны предметы, несмотря на то, что

лучи света, как правило, преломляются, т. е. отклоняются от первоначального направления. Диффузное пропускание света происходит сквозь просвечивающиеся материалы (замерзшее стекло), в которых поверхностные неоднородности приводят к беспорядочному перемешиванию световых лучей. Поэтому очертания предмета, рассмотренного через такие материалы, размыты.

### Литература

1. Авдеева, С.М. Алгоритмы трехмерной машинной графики: учеб. пособие / С.М. Авдеева, С.М. Куров. – Москва : МГТУ им. Н.Э. Баумана, 1996. – 60 с.: ил.
2. Никулин, Е. А. Компьютерная геометрия и алгоритмы машинной графики / Е.А. Никулин. – Санкт-Петербург : БХВ-Петербург, 2003. – 560 с.: ил.
3. Порев В.Н. Компьютерная графика / В.Н. Порев. – Санкт-Петербург : БХВ-Петербург, 2002. – 432 с.: ил.
4. Тихомиров, Ю. Программирование трехмерной графики / Ю. Тихомиров. – Санкт-Петербург : ИРМ - Санкт-Петербург, 1998.
5. Эйнджел, Э. Интерактивная компьютерная графика / Э. Эйнджел. – Москва : «Вильямс», 2001. – 592 с.: ил.

**К.В. Попков (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Кулинченко**, ст. преподаватель

### ПРИМЕНЕНИЕ УТИЛИТЫ SUDO

Linux – общее название для UNIX-подобных ОС, основанных на ядре с таким же названием. В настоящее время Linux занимает лидирующие места на рынке смартфонов, суперкомпьютеров, интернет-серверов и половину встраиваемых систем. На рынке домашних компьютеров Linux не занимает большую долю, хотя энтузиастов хватает. И для этих энтузиастов создано огромное количество дистрибутивов от самых простых, работающих из коробки (Ubuntu, Mint), до дистрибутивов, созданных для опытных пользователей (Gentoo, ArchLinux). Linux отличается от других открытым исходным кодом, высоким уровнем безопасности, надежности и свободным распространением.

В любой системе Linux обязательно есть один привилегированный пользователь – root. Суперпользователь имеет, как правило, все права и доступ ко всем материалам. Обычные пользователи под этой ОС не имеют права практически ни на что. Даже копирование и перемещение,

не обладая правами суперпользователя, выполнить не удастся. Поэтому в дистрибутивах Linux использовалась утилита su. Её задача состояла в том, чтобы дать права root любому пользователю в системе. То есть нужно было ввести пароль суперпользователя и если пароль был верным она подменяла текущее имя пользователя на root. Все пользователи состоявшие доверенной группе «wheel» знали пароль root'a и обладали неограниченным доступом к системе. Это являлось серьезной проблемой безопасности. Позже появилась утилита sudo и своими возможностями шагнула далеко вперед.

Sudo – это утилита, предоставляющая привилегии root для выполнения административных операций в соответствии со своими настройками. Она предоставляет контроль к приложениям системы, т. е. фактически любой пользователь может обладать теми же возможностями, что и root. Однако такое поведение Sudo легко контролировать, изменять и настраивать. Sudo используется всегда, когда запускается что-то из меню «Администрирование системы» или из терминала.

Однако не все программы запускаются через Sudo. Многие программы нужно запускать вручную.

Для примера использования Sudo можно выбрать, наверное, самую первую и популярную команду терминала. Обновление системы:

```
Sudo apt-get update
```

Происходит обновление индекса пакетов до последних изменений.

```
Sudo apt-get upgrade
```

Обновление системы.

Или, к примеру, удаление какого-нибудь приложения:

```
Sudo apt-get remove Имя приложения
```

Терминал запросит пароль суперпользователя, при его вводе система выполнит требуемые пользователем действия. Без команды Sudo ничего не отработает.

Важным преимуществом этой утилиты является возможность установления ограничений для пользователей. В этом можно убедиться на простом примере: имеется группа с названием «Group», состоящая из пользователя «User». Эта группа может с помощью sudo выполнять команду /usr/local/bin/mis\_ext \*, звездочкой являются передаваемые в сценарий параметры. Но пользователю «User» будет запрещен этот сценарий с параметром import. Такие ограничения задаются с помощью значка '!' – логического оператора НЕ. Вот что указывается в файле sudoers в конечном итоге:

```
%Group rs6000 = (dbmis) NOPASSWD: /usr/local/bin/mis.ext *
```

```
User rs6000 = (dbmis) NOPASSWD: !/usr/local/bin/mis_ext import
```

У Sudo есть еще одна очень интересная возможность. Так называемый тайм-аут. Она заключается в том, что пользователь вводит пароль (свой собственный) и в течении определенного времени может его не вводить повторно каждый раз, когда выполняет какие-нибудь команды. По умолчанию этот промежуток времени равен пяти минутам. Его можно изменить, добавив запись в файл sudoers. Смена производится для конкретного пользователя, в данной ситуации для пользователя «User»:

```
Defaults: user timestamp_timeout=20
```

Теперь тайм-аут для пользователя «User» будет составлять 20 минут. Чтобы Sudo всегда требовал ввод пароля, следует установить значение, равное 0. Чтобы срок действия пароля не истекал, нужно установить любое отрицательное число.

Чтобы разрешить пользователю запускать определенный набор команд без ввода какого-либо пароля, нужно начать с команды NOPASSWD: :

```
User localhost = NOPASSWD: /usr/bin/emerge
```

Для работы в широком окружении существует такое свойство, как использование псевдонимов, т. к. необходимость ввода пользователей снова и снова будет долгой и неинтересной задачей. Эта операция проводится с целью облегчить администрирование. Псевдонимы (aliases) объявляются так:

```
Host_Alias hostalias = hostname1, hostname2, ...
```

```
User_Alias useralias = user1, user2, ...
```

```
Cmnd_Alias cmndalias = command1, command2, ...
```

Есть псевдоним, который работает всегда и везде. Это псевдоним ALL. С помощью этого примера можно разрешить действия всем пользователям, например, выполнение команды shutdown:

```
ALL localhost = /sbin/shutdown
```

Ранее упомянутый файл настроек /etc/sudoers всегда следует редактировать с помощью команды visudo. Visudo блокирует файл sudoers, сохраняет изменения во временный файл и проверяет, что файл грамматически корректен перед тем, как скопировать его в /etc/sudoers. Редактируя файл таким способом, можно быть уверенным в том, что ошибок в файле не будет. Любая ошибка делает sudo неработоспособным.

Говоря на простом языке, Sudo – это утилита, дающая право на контроль за тем, кто и как использует команды и привилегии в системе. С помощью этой утилиты можно настраивать права на использование для обычных пользователей, вводя любые ограничения.

Аналогом данной утилиты в Windows является команда RunAs. Она запускает конкретные средства и программы с разрешениями, отличными от тех, которые предоставляет текущая учетная запись.

## **НОВЫЕ ВОЗМОЖНОСТИ 1С:ПРЕДПРИЯТИЯ ВЕРСИИ 8.2**

Версия 1С:Предприятие 8.2 – это принципиальное изменение архитектуры платформы версии 8, наиболее существенное с момента ее выпуска. В ней реализовано много новых механизмов и новых подходов к разработке.

1С:Предприятие 8.2 полностью меняет весь слой работы с интерфейсом. Сюда относятся и командный интерфейс, и формы, и оконная система. При этом не только меняется модель разработки пользовательского интерфейса в конфигурации, но и предлагается новая архитектура разделения функциональности между клиентским приложением и сервером.

### **Основные изменения:**

#### *1 Несколько дистрибутивов*

Теперь существует два дистрибутива: «1С:Предприятие 8.2» и «1С:Предприятие 8.2 Тонкий клиент». Дистрибутив «1С:Предприятие 8.2» – это как и раньше, полный дистрибутив платформы, который позволяет использовать все ее возможности. Дистрибутив «1С:Предприятие 8.2 Тонкий клиент» – это только клиентское приложение. Основным сценарий его использования – это работа с клиент-серверной базой по локальной сети или по протоколу НТТР. Возможность работы с файловой информационной базой непосредственно по сети отсутствует у тонкого клиента, установленного из этого дистрибутива.

#### *2 Установка, обновление и запуск*

Новый механизм позволяет значительно упростить администрирование системы в тех случаях, когда одновременно используются несколько разных версий платформы. С его помощью автоматизируется их установка, выбор подходящей версии платформы и подходящего клиентского приложения для запуска, а также процесс своевременного обновления платформы.

#### *3 Каталог шаблонов конфигураций и обновлений*

Раньше существовал единственный локальный каталог шаблонов конфигураций и обновлений. При создании новой информационной базы пользователю представлялась возможность создать ее на основе одного из шаблонов, находящихся в этом каталоге. Теперь механизм шаблонов конфигураций сохранился, но перетерпел некоторые изменения:

– может существовать сразу несколько каталогов шаблонов конфигурации и обновлений;

– изменено место хранения путей к этим каталогам.

#### 4 Конвертация

Чтобы информационную базу версии 8.1 запустить под новой версией платформы, ее нужно конвертировать. После конвертации не требуется дополнительная доработка.

#### 5 Конфигурацию можно переделывать частями

Версия платформы 8.2 позволяет работать с разными частями конфигурации одновременно и в старом, обычном режиме, и в новом режиме, режиме управляемого приложения.

#### 6 Новый интерфейс

Работа конфигурации в режиме управляемого приложения визуально отличается от старого интерфейса 1С:Предприятия 8.1 в этом режиме пользователь взаимодействует с управляемым интерфейсом и управляемыми формами.

При создании платформы 1С:Предприятия 8.2 была проделана большая работа по разработке нового современного интерфейса.

#### 7 Три клиентских приложения

В версии 1С:Предприятие 8.2, три клиентских приложения (рисунок 1) вместо одного в версии 8.1:

- толстый клиент (старый) – 1cv8.exe;
- тонкий клиент – 1cv8c.exe;
- веб-клиент.

Все эти программы, за исключением веб- клиента, находятся в каталоге исполняемых файлов конкретной версии платформы.

	Толстый клиент	Тонкий клиент	Веб-клиент
Разработка прикладных решений	Да	Нет	Нет
Работа в локальной сети	Да	Да	Да
Работа через интернет	Нет	Да	Да
Необходимость предварительной установки	Да, большой дистрибутив	Да, маленький дистрибутив	Нет

Рисунок 1 – Типы клиентов

#### 8 Режим низкой скорости соединения

Предназначен для случаев, когда работа выполняется по протоколу HTTP через низкоскоростное Интернет-соединение (например, GPRS).

#### 9 Многоплатформенность

В версии 1С:Предприятие 8.2, благодаря появлению веб-клиента, все компоненты системы могут работать на компьютерах как под управлением различных операционных систем.

#### 10 Отказоустойчивый кластер



Масштабируемость. Можно управлять распределением нагрузки, которая раньше приходилась на единственный менеджер кластера. Теперь эта нагрузка может быть распределена между несколькими менеджерами кластера, что позволяет разгрузить главный менеджер кластера и повысить надежность его работы.

Отказоустойчивость кластера в целом достигается за счет того, что в 1С:Предприятии 8.2 несколько кластеров могут быть объединены в группу резервирования. Кластеры, находящиеся в одной группе резервирования синхронизируются автоматически.

Динамическое распределение нагрузки. Загруженность рабочих процессов анализируется динамически и при необходимости клиент автоматически переключается на более производительный рабочий процесс. Такое переключение происходит незаметно для пользователя.

#### 11 *Oracle DataBase*

В 1С:Предприятии 8.2 реализована поддержка новой СУБД – Oracle Database. Теперь платформа 1С:Предприятия 8 поддерживает работу с пятью СУБД:

- файловая;
- Microsoft SQL Server;
- PostgreSQL;
- IBM DB2;
- Oracle Database.

#### 12 *С файловой базой можно работать не всегда*

В версии 8.2 есть две ситуации, в которых не поддерживается работа системы 1С: Предприятие в файловом режиме работы:

- когда тонкий клиент или веб клиент работают с базой данных через веб-сервер.
- это создание начального образа клиент-серверной информационной базы.

#### 13 *Windows 98/Me больше не поддерживается*

В версии 1С: Предприятия 8.2 все компоненты платформы являются полностью Unicode версий API. Следствием этого является невозможность работы компонентов платформы на не Unicode версиях Windows, Таких как Windows 98 и Windows Me.

#### 14 *Новая технология внешних компонентов*

Существовавшая ранее технология внешних компонентов была основана на технологии COM. По этой причине внешние компоненты могли работать только в операционных системах Microsoft Windows. Для 1С:Предприятия 8.2 разработана новая технология внешних компонентов. Теперь основной технологией, которая используется для создания внешних компонентов, является технология Native API – собственный интерфейс системного программирования 1С: Предприятия 8.

## **ОСОБЕННОСТИ КЛИЕНТОВ 1С:ПРЕДПРИЯТИЯ ВЕРСИИ 8.2**

1С:Предприятие 8.2 содержит множество нововведений и изменений, не всегда заметных с первого взгляда. В версии 1С:Предприятие 8.1 было одно клиентское приложение – 1cv8.exe. Это программа, которая работала на компьютере пользователя и обеспечивала интерактивное взаимодействие системы 1С:Предприятие 8 с пользователем. В версии 1С:Предприятие 8.2, клиентских приложения (рисунок 1) три вместо одного.

	<i>Толстый клиент</i>	<i>Тонкий клиент</i>	<i>Веб-клиент</i>
<i>Разработка прикладных решений</i>	<i>Да</i>	<i>Нет</i>	<i>Нет</i>
<i>Работа в локальной сети</i>	<i>Да</i>	<i>Да</i>	<i>Да</i>
<i>Работа через интернет</i>	<i>Нет</i>	<i>Да</i>	<i>Да</i>
<i>Необходимость предварительной установки</i>	<i>Да, большой дистрибутив</i>	<i>Да, маленький дистрибутив</i>	<i>Нет</i>

Рисунок 1 – Типы клиентов 1С: Предприятие 8.2

Выделяют следующие типы клиентов:

- толстый клиент (старый) – 1cv8.exe;
- тонкий клиент – 1cv8c.exe;
- веб-клиент.

Все эти программы, за исключением веб-клиента, находятся в каталоге исполняемых файлов конкретной версии платформы.

Толстый клиент позволяет реализовывать полные возможности 1С:Предприятия 8.2 как в плане разработки, администрирования, так и в плане исполнения прикладного кода. Однако он не поддерживает работу с информационными базами через Интернет, требует предварительной установки на компьютер пользователя и имеет большой объем дистрибутива.

Тонкий клиент не позволяет разрабатывать и администрировать прикладные решения, однако может работать с информационными базами через Интернет. Он также требует предварительной установки на компьютер пользователя, но имеет меньший размер дистрибутива, чем толстый клиент.

Веб-клиент не требует какой-либо предварительной установки на компьютер. В отличие от толстого и тонкого клиентов, он исполняется не в среде операционной системы компьютера, а в среде интернет-браузера (Microsoft Internet Explorer или Mozilla Firefox). Поэтому

пользователю достаточно всего лишь запустить свой браузер, ввести адрес веб-сервера, на котором опубликована информационная база – и веб-клиент «сам придет» к нему на компьютер и начнет исполняться.

**П.М. Пранович (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **Е.А. Левчук**, канд. техн. наук, доцент

## **АНАЛИЗ И ПОИСК РЕШЕНИЙ МОДЕРНИЗАЦИИ ЛВС В МОГИЛЁВСКОМ ФИЛИАЛЕ РУП «БЕЛТЕЛЕКОМ» БОБРУЙСКИЙ УЗЕЛ ЭЛЕКТРОСВЯЗИ**

Развитие общества в техническом плане обусловлено предоставлением различными компаниями широкому кругу пользователей инновационных технологий. Это утверждение справедливо и в отношении РУП «Белтелеком», ведь без модернизации своего оборудования можно создать лишь видимость инноваций. На сегодняшний день РУП «Белтелеком» предоставляет такие услуги как высокоскоростной доступ в Интернет, коммутация частных виртуальных сетей (VPN), IP-телевидение, конечно же телефонную связь и много другое.

Локальная сеть Могилёвского филиала РУП «Белтелеком» Бобруйского узла электросвязи охватывает четыре этажа здания. Кабельная система сети имеет комбинированную топологию и построена на основе кабеля UTP категории 5е и многомодового волоконно-оптического кабеля, что позволяет передавать огромные массивы информации на высоких скоростях. Однако, объемы передаваемой информации растут с каждым днем.

Основные задачи и требования, возлагаемые на оборудование ЛВС Могилёвского филиала РУП «Белтелеком» Бобруйского узла электросвязи:

- предоставление населению и организациям качественного канала связи для доступа в сеть Интернет;
- коммутация и разграничение сетевого трафика (телефония, телевидение, данные и т. д.);
- объединение собственного компьютерного парка в локальную вычислительную сеть;
- коммутация частных виртуальных сетей;
- коммутация WEB серверов;
- высокая надежность и отказоустойчивость сети в целом.

Локальная вычислительная сеть здания Могилёвского филиала РУП «Белтелеком» Бобруйского узла электросвязи содержит 6 коммутаторов

HP ProCurve switch 2626 (J4900B). Выбор сетевого оборудования компании HP обусловлен курсом РУП «Белтелеком» на качество предоставляемых услуг. Возможности коммутаторов HP серии ProCurve Switch 2600 позволят выгодно развивать бизнес и предлагать заказчикам более защищенные, производительные и гибкие решения. Пожизненная гарантия, быстрые сроки поставки и легендарный сервис HP полностью удовлетворяют требования РУП «Белтелеком».

Каналы коммутаторов имеют несколько резервных линий. Для обеспечения безопасности внутренней локальной вычислительной сети применяется статическая IP-адресация, внешняя же адресация динамическая. В качестве серверного оборудования были выбраны серверы компании HP. Доступ в глобальную сеть и связь с остальными объектами РУП «Белтелеком» реализован посредством технологии GPON.

Из анализа локальной вычислительной сети Могилёвского филиала РУП «Белтелеком» Бобруйского узла электросвязи сделаны следующие выводы:

- в сети филиала установлено современное оборудование, модернизация которого не имеет смысла;
- необходимо сконфигурировать межсегментные соединения коммутаторами различного уровня для повышения надёжности работы сети;
- для повышения пропускной способности необходимо использование специального коммутационного оборудования, применение алгоритмов сжатия информации и/или протоколов параллельной передачи данных на конечных устройствах.

**В.И. Пышненко (УО «БГУИР», Минск)**

Науч. рук. **С.Н. Неборский**, канд. физ.-мат. наук, доцент

## **МОДЕЛЬ КАЧЕСТВА СЕТЕЙ ГОЛОСОВОЙ СВЯЗИ НА ОСНОВЕ ОБОРУДОВАНИЯ CISCO**

С ростом комплексных потребностей в области интегрированных услуг по передаче голоса качество обслуживания (QoS), стало ключевым фактором сервисов с жесткими гарантиями качества обслуживания.

VoIP (Voice over IP) стремительно изменяет облик современной телефонии. Термин «Voice over IP» подразумевает под собой VoIP-сети, включая потоковые и сигнальные протоколы, а также кодеки.

Для оценки качества программных средств VoIP предлагается следующая модель качества:

$$Q = \{F, R, E, M\},$$

где  $F$  – множество подхарактеристик функциональности,  
 $R$  – множество подхарактеристик надежности,  
 $E$  – множество подхарактеристик эффективности,  
 $M$  – множество подхарактеристик сопровождаемости.

Данная модель основана на стандарте ISO/IEC 25010:2011 Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models [1].

Для внедрения VoIP-решений требуется тщательный анализ сетевых требований и текущих условий с целью обеспечения качества звонка, сопоставимого с аналоговой средой (PSTN – Public Switched Telephone Network). Cisco является одной из компаний, работающих в области создания технологий и протоколов, повышающих качество передачи голоса в сетях IP.

Определяя требования функций качества обслуживания корпоративного трафика IP-телефонии, рекомендуется придерживаться следующих правил:

- Голосовой трафик должен быть промаркирован как DSCP EF, в соответствии с «Базовыми Основами QoS» и RFC 3246.

- Сигнализация должна быть промаркирована как CS3, в соответствии с «Базовыми Основами QoS» (во время миграции можно использовать AF31).

- Потери пакетов в магистральных спроектированных для предоставления VoIP сервиса высокого качества не должны превышать 0,25 %.

- Односторонняя задержка не должна превышать 150 ms, в соответствии со International Telecommunication Union (ITU – Международный Союз Электросвязи) G.114.

- Колебания задержки (jitter) должны быть менее 10 мсек.

- Для каждого разговора (в зависимости от частоты квантирования, кодека и заголовка второго уровня) требуется 21–106 kbps гарантированной приоритетной полосы пропускания.

- Для трафика сигнализации требуется 150 bps (плюс заголовков второго уровня) гарантированной полосы пропускания.

На качество голосовой связи напрямую влияют все три фактора качества QoS:

- Потери пакетов;
- Задержка;
- Вариации задержки.

Также для обеспечения QoS в Cisco IOS поддерживаются следующие протоколы [2]:

- WFQ – протокол взвешенной справедливой очередности (Weighted Fair Queuing) .

- Протокол приоритетной очередности и обычной очередности (Priority Queuing and Custom Queuing).
- WRED – протокол взвешенного случайного раннего обнаружения (Weighted Random Early Detection).
- Коммутация по меткам (Tag Switching) / многопротокольная коммутация по меткам (MPLS)

Потери пакетов вызывают кратковременные пробелы в разговоре. Стандартные алгоритмы кодирования, используемые в Cisco Digital Signal Processor (DSP), с помощью алгоритмов маскирования могут восстановить потери до 30 мсек. Таким образом, потери двух и более последовательных 20 мсек сэмплов приведут к заметной деградации качества голоса.

В стандарте Международного Союза Электросвязи для технологии VoIP (G.114) говорится, что задержка величиной в 150 мсек в одном направлении является приемлемой для качество голосовой связи. Было продемонстрировано, что разница в качестве голоса между сетями с задержкой в 150 мсек и 200 мсек является незначительной и практически незаметной для пользователя. Cisco рекомендует ориентироваться на ITU стандарт 150 мсек, но если существуют ограничения не позволяющие добиться такого бюджета, то размер задержки может быть увеличен до 200 мсек без значительной деградации качества связи.

Что же касается колебаний задержки, то для их выравнивания в устройствах Cisco для IP-телефонии используются адаптивные буферы. Однако они могут компенсировать колебания задержки лишь в пределах от 20 до 50 мсек. При централизованной обработке вызовов IP-телефоны используют контрольные каналы TCP для связи с Cisco CallManager.

Качество голосовой связи во многом зависит от обработки голоса на шлюзе. Шлюз Cisco поддерживает множество средств кодирования и декодирования (CODEC). Для обеспечения высокой устойчивости к неблагоприятным условиям окружающей среды и шлюзы могут устанавливаться на платформах, отвечающих стандарту NEBS (Network Equipment-Building System – Система создания сетевого оборудования). Этот стандарт используется региональными операторами (Regional Bell Operating Companies – RBOC), чтобы гарантировать устойчивость оборудования к экстремальным температурам, высокой влажности, условиям высокогорья, пожарам, землетрясениям, повышенной вибрации, всплескам напряжения в сети питания, прекращению подачи электричества и электромагнитным помехам.

При проектировании сети операторы должны учитывать требования к качеству обслуживания, предъявляемые корпоративными клиентами. Общая проблема операторов и корпоративных клиентов заключается

в богатстве QoS функциональности Cisco iOS и, как следствие, мириады вариантов реализации и комбинаций. Практически каждый опытный инженер имеет свой собственный взгляд на их применение. Для того, чтобы дать некоторые общие рекомендации по реализации качества обслуживания, Cisco воплотила новую инициативу, под названием «Базовые Основы QoS», целью которой является унификация решений на платформах оборудования Cisco. В «Базовых Основах QoS» [3] специфицирована маркировка и правила обработки до 11 классов сервиса в корпоративных сетях. Важно отметить, что «Базовые Основы QoS» не диктуют каждому корпоративному клиенту немедленно внедрить 11 классов трафика, а скорее учитывают существующие и будущие потребности в поддержке QoS. Даже если корпоративному клиенту сейчас нужна только часть из этих 11 классов, то следование рекомендациям «Базовых Основ QoS» позволит им в будущем плавно мигрировать на расширение количества поддерживаемых классов в будущем.

### **Литература**

1. ISO/IEC 25010:2011 Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models. – 2011.
2. Yen, Y.S, Nat. Dong Hwa, Chen, W. Sliding weighted fair queueing scheme for real-time applications// Communications, IEE Proceedings. – 2005.–№152. – P. 320–326.
3. How QoS Mechanisms Affect VoIP QoS Metrics// IDE1070. – 2010.

**В.И. Рагин (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **А.В. Воружев**, канд. техн. наук, доцент

### **СРАВНЕНИЕ И РЕАЛИЗАЦИЯ ТЕХНОЛОГИЙ ЗАГРУЗКИ КЛИЕНТСКИХ СТАНЦИЙ**

Сегодня на рынке представлено несколько решений создания рабочего места клиента, из них: локальная загрузка, удаленная загрузка тонкого клиента, и бездисковая загрузка.

На данном этапе, мало представлена бездисковая загрузка, однако, на определенные типы задач, работа с графикой, звуком, обучение студентов, она является наиболее выгодной, как экономически, так и с точки зрения безопасности.

Для аргументации использования нового метода загрузки, нам необходимо сравнить, насколько различается взаимодействие пользователей

с локальной и удаленной машиной. В качестве наглядного материала, мы будем использовать UML диаграмму процессов.

На рисунке 1 показан процесс взаимодействия с ПК пользователя локальной системы. Это классическая система работы и предполагаемый переход на удаленную загрузку является попыткой оптимизации ресурсов и упрощению администрирования системы в целом. Основным узким местом классической системы являясь, в первую очередь, безопасность. Не смотря на развитие антивирусных программ, дать максимальную гарантию защиты позволяют решения, которые принципиально запрещают вносить изменение. Это преимущество хорошо реализовано в удаленном варианте загрузки.

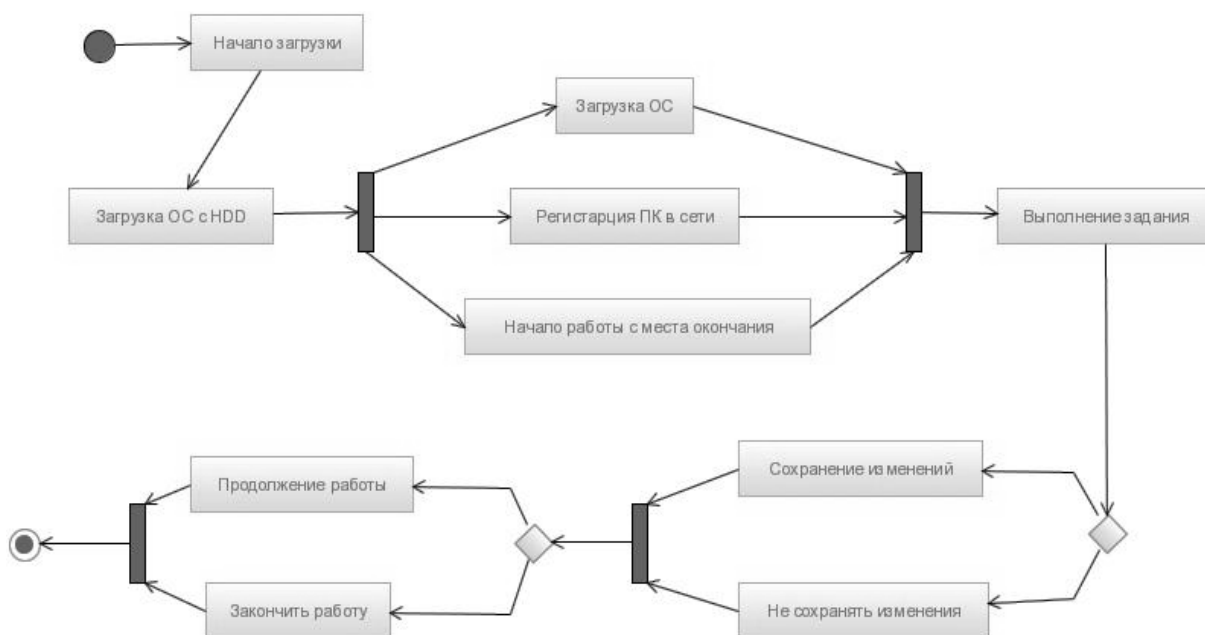


Рисунок 1 – UML-диаграмма процесса локальной загрузки

Рисунок 2 показывает ту же самую работу, но с использованием удаленной загрузки. Как видно из сравнения, принципиальных отличий не существует. Пользователи одинаково используют ПК. Для пользователя, разница заключается лишь в невозможности сохранить результат изменения системы.

Т. е. несмотря на действия защиты методом запрета, у пользователя остаётся широкий круг возможностей. Раньше, при реализации политики доменов в локальных машинах, требовалось ограничивать пользователей в административных правах. В данном случае этот пункт является условным. Никакие изменения, сделанные в машине не сохраняются при последующей перезагрузке. Ограничение прав администратора имеет смысл лишь в повышении мер безопасности на сетевом уровне.



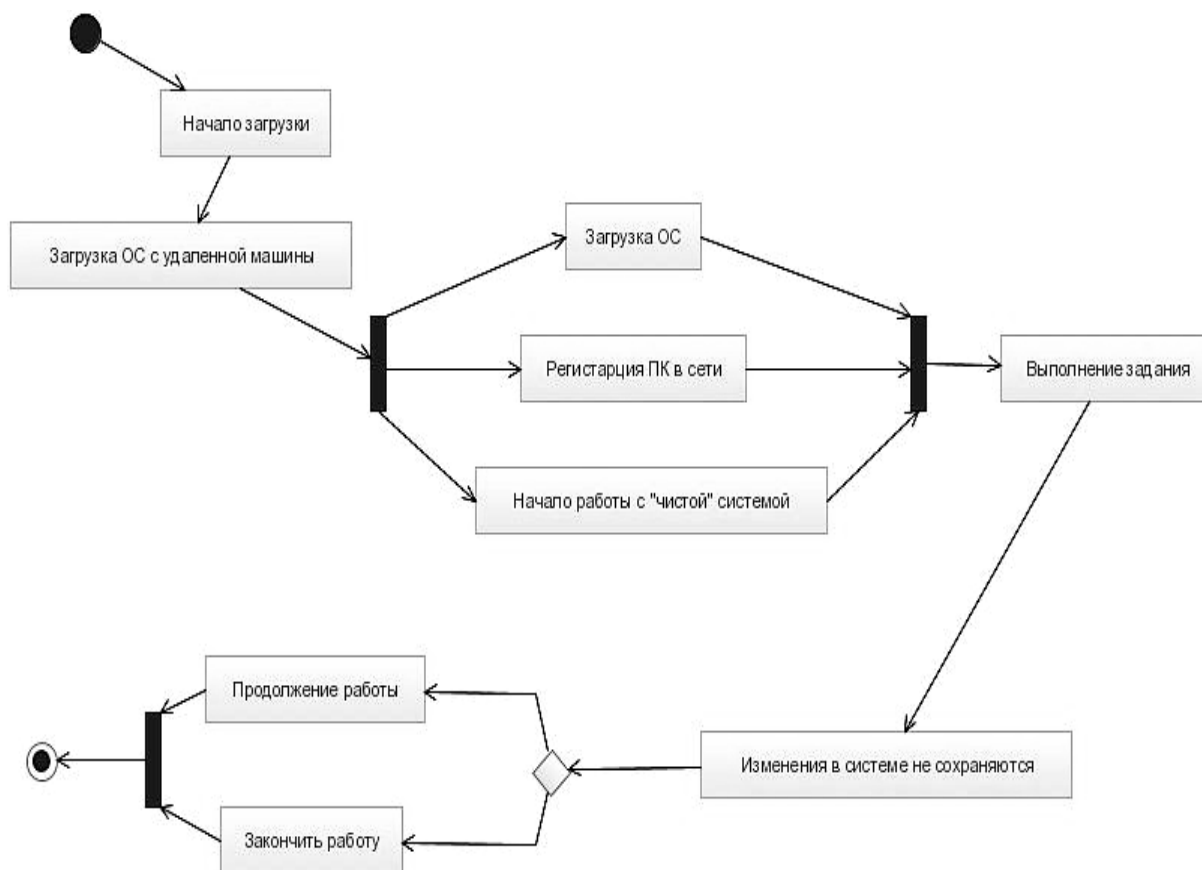


Рисунок 2 – UML-диаграмма процесса удаленной загрузки

Основой смысл использования удаленной загрузки – это формат работы как с локальной машиной, но с использованием преимуществ тонкого клиента.

Исходя из теории и практик, тонкий клиент (рисунок 3), передает выполнение задач на сервер и не требователен к машине клиента, но в данном случае невозможно полноценно вести работу с высоконагруженными приложениями, такие как обработка графики.

Работая с тонким клиентом, путь информационного обмена значительно увеличивается. Что видно из диаграммы. Такой метод применим в случае, как уже писалось выше, низких нагрузках единичного пользователя и при необходимости развернуть большое количество дешевых рабочих мест.

Тонкий клиент хорошо подходит для использования в бухгалтерии, офисах и т. п., где нет претензий к времени обработки и прямого соединения с оборудованием. В рассматриваемой модели университета, требуется обработать информацию графического содержания и иметь возможность подключения оборудования физически к учебным маршрутизаторам, проекторам и т. п. Вследствие этого, использование тонкого клиента не является целесообразным в данном случае.

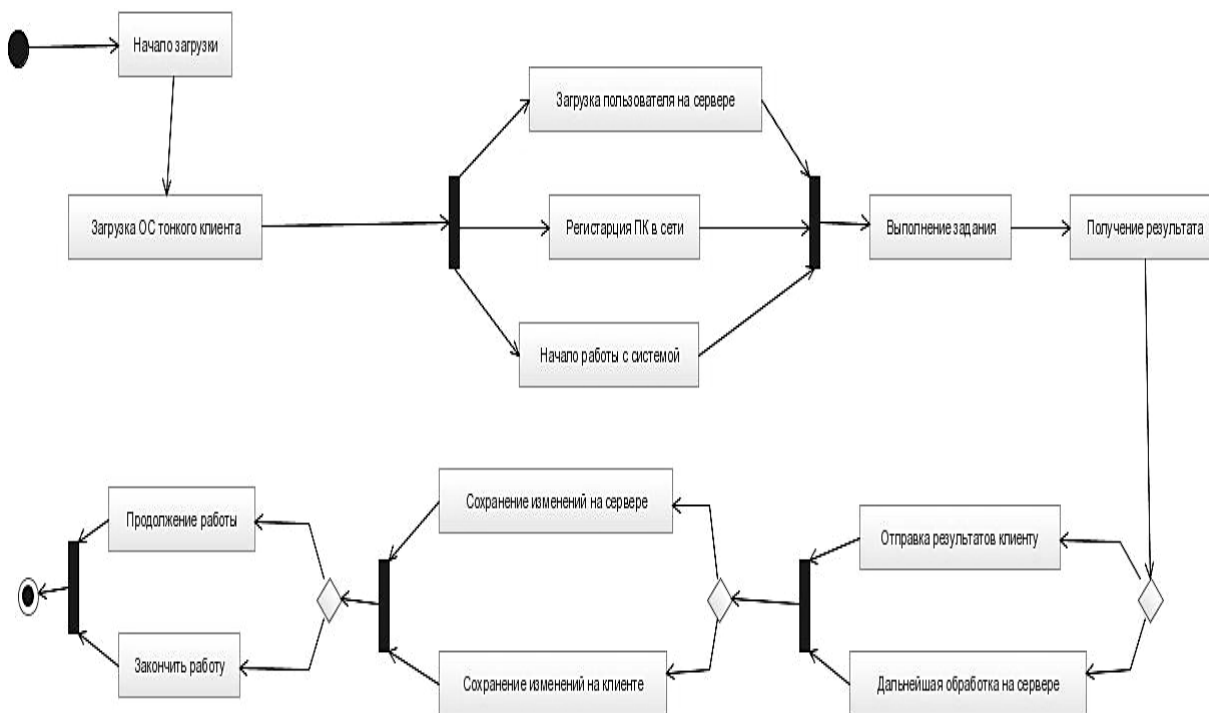


Рисунок 3 – UML-диаграмма процесса загрузки тонкого клиента

## Литература

1. Официальный сайт программы управления данными Starwind [Электронный ресурс]. – 2014. – Режим доступа: <http://ru.starwind-software.com/>. – Дата доступа: 25.11.2014.
2. Коллективный блог [Электронный ресурс]. – 2014. – Режим доступа: <http://habrahabr.ru/>. – Дата доступа: 25.12.2014.
3. Справка по продуктам Microsoft [Электронный ресурс]. – 2014. – Режим доступа: <http://technet.microsoft.com/>. – Дата доступа: 25.10.2014.
4. Моримото, Р. Microsoft Windows Server 2012. Полное руководство / Р. Моримото, М. Ноэл. – М.: ООО «Вильямс», 2013. – 1456 с.: ил.

**В.И. Рагин (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **А.В. Воружев**, канд. техн. наук, доцент

## **ПРОЕКТ БЕЗДИСКОВОЙ ЗАГРУЗКИ КЛИЕНТСКИХ СТАНЦИЙ В СЕГМЕНТЕ ЛВС КАФЕДРЫ АСОИ**

Любое современное учреждение нуждается в оснащении рабочих мест компьютерами. Чаще всего не требуется вносить изменения в систему во время обычной работы пользователя, с помощью технологии iSCSI образ системы может быть загружен на определенном количестве

клиентов и, после завершения работы, система не сохранит сделанных изменений, если это отдельно не предусмотрено.

Следовательно, в настраиваемом образе не требуется антивируса, достаточно установить только Firewall для защиты сети от активного заражения.

Для реализации проекта, был выбран StarWind iSCSI SAN с бесплатной лицензией. За серверную платформу был взят MS Server 2012, бесплатный для студентов, преподавателей и учебных заведений. MS Server включает в себя большинство необходимых служб и настроек:

- встроенный TFTP сервер, входящий в компонент служб развертки;
- DHCP и NAT. При настройке, Особое внимание необходимо обратить на дополнительные параметры DHCP, параметры 66, 67, с их помощью мы задаем имя загружаемого файла и IP сервера TFTP;
- DNS и Active Directory для соединения с сетевыми учетными записями, если данные службы не активированы во внешней сети.

Образ системы, устанавливается на виртуальной машине и настраивается, обязательна установка CCboot инициатора, для работы с StarWind iSCSI SAN.

По завершению настройки, конвертируем систему в img образ с помощью StarWind V2V Image Converter. Полученный образ добавляем в StarWind iSCSI SAN и настраиваем таргет. Раздача образов будет определяться MAC адресом машины.

Для настройки машины, необходимо в основных настройках BIOS сетевой карты клиента выставить первичную загрузку.

Подключенный образ можно настроить как на запись, так и на отклонения сделанных изменений в системе. Во время первой загрузки, необходимо обновить драйвера и ПО, а так же произвести персональную настройку рабочего места. Сделанные изменения сохраняются в файле snapshot и могут быть использованы при следующей загрузке системы, не внося изменения в основной образ. После настройки, файл snapshot так же блокируется для записи.

**А. А. Ракицкий (УО «ГГТУ им. П.О. Сухого», Гомель)**  
Науч. рук. **Т.В. Тихоненко**, канд. физ.-мат. наук

## **ТЕСТИРОВАНИЕ АНТИВИРУСНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

В настоящее время всё чаще и чаще появляются новые версии и модификации компьютерных вирусов, вред от которых всё более и серьезнее.

В данной работе изложена основная информация о компьютерных вирусах и о программах, которые помогают бороться с ними.

Для того, чтобы не быть голословным, автор собрал всю информацию об нескольких, наиболее популярных и недавно появившихся на просторах стран СНГ, антивирусных решениях, а также, провел полное тестирование этих антивирусных решений, в котором проверялись основные функции, которыми должен обладать современный антивирусный продукт.

В тестировании участвовали следующие решения:

- Avast! Antivirus Free 2014 9.0.2011 – бесплатный
- Bitdefender Antivirus Free Edition 1.0.21.1099 – бесплатный
- Dr.Web Security Space 9.0.0.12100 – платный
- ESET NOD32 Smart Security 7 v.7.0.302.26 – платный
- Kaspersky Internet Security 2014 14.0.0.4651 (d) – платный
- Qihoo 360 Internet Security 2014 4.9.0.4900 – бесплатный

Тестирование проводилось по следующим критериям:

– *Обнаружение* – общий уровень обнаружения антивируса (количество обнаруженных угроз) в процентах при проверке 3039 вредоносных файлов.

– *Динамика* – процент угроз нейтрализованных при запуске от оставшихся нераспознанных вредоносных файлов после сканирования.

– *Веб-защита* – процент заблокированных фишинговых и вредоносных веб-сайтов. Использовался набор из 10 фишинговых и 5 вредоносных веб-сайтов.

– *Самозащита* – уровень самозащиты антивируса при проверке с помощью Process Hacker выгрузки основных работающих в системе процессов антивирусной защиты с последующим контролем защитных возможностей.

– *Ложные* – количество ложных срабатываний при проверке 855 709 чистых файлов после установки 150 программ в систему.

Исходя из полученных результатов, была составлена специальная формула, по которой давалась общая оценка продукта.

Формула:

$$\begin{aligned} & (\text{Обнаружение} * 1,5 + \text{Динамика} * 1,5 + \text{Веб-защита} * 1 + \\ & + \text{Самозащита} * 0,5 + \text{Ложные} * 0,5) / 5. \end{aligned}$$

В зависимости от важности каждой функции, процент умножался на её приоритет: 1,5–0,5. Затем сумма этих баллов делилась на количество критериев тестирования.

Условия проведения тестирования:

1. В тестировании принимают участие антивирусные решения, официально совместимые с системой Windows 8.1 на момент начала тестирования.

2. В тестировании участвуют продукты класса Internet Security и бесплатные антивирусы, если такие решения есть у вендора.

3. Режим работы операционной системы – административная учетная запись с отключением всех дополнений защиты от Microsoft. Исключением было тестирование операционной системы с установленным Защитником Windows (Windows Defender), где необходимо было проверить защиту непосредственно инструментов защиты Windows, при этом использовалась пользовательская учетная запись. Для сторонних антивирусных программ использование административной учетной записи обусловлено тем, что проверялась работа непосредственно защитного продукта без дополнительных средств защиты самой операционной системы.

4. Тест проводится с настройками по умолчанию за исключением изменения возможности принятия решения по обнаруженным угрозам, так как у некоторых продуктов настроено обращение к пользователю для принятия решения при каждом обнаружении.

5. Проверка вылеченных файлов производится путем перепроверки остатков дополнительными антивирусными сканерами и на VirusTotal, которые изначально подтверждают угрозу, а после обработки антивирусом – не подтверждают. Разница в показаниях и есть количество вылеченных файлов.

6. Для определения нагрузки на процессор и основной работающий файл используется COMODO KillSwitch, так как по ряду причин эта программа является максимально подходящей для подобного тестирования.

7. Для создания образов операционных систем с установленными антивирусами используется программа O&O DiskImage Professional 7.2.10.

8. Для тестов использовались следующие образцы и инструменты:  
– 3039 вредоносных файлов для проверки сканирования по требованию;

– нераспознанные файлы для запуска в динамическом тестировании;

– набор из 10 фишинговых и 5 вредоносных веб-сайтов для проверки веб-защиты.

Результаты тестирования можно увидеть на рисунке 1.

Подводя итоги, можно сказать следующее:

– абсолютно идеального антивирусного продукта не существует – у каждого есть свои достоинства и свои недостатки;

– результаты тестирования нельзя назвать абсолютно безоговорочными, поскольку при проведении тестирования возникают разного рода нюансы, которые тоже стоит учитывать, а некоторые просто не предугадаешь.


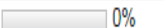
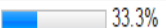
Антивирусные программы	Обнаружение	Динамика	Веб-защита	Самозащита	Ложные	Оценка
Qihoo 360 Internet Security 2014	 98.1%	 100%	 100%	 50%	 3	9.2
Kaspersky Internet Security 2014	 92.4%	 87.5%	 86.7%	 100%	 0	9.1
Bitdefender Antivirus Free Edition	 96.7%	 75%	 100%	 50%	 0	8.7
ESET NOD32 Smart Security 7	 95.2%	 37.5%	 86.7%	 50%	 0	7.2
Avast! Antivirus Free 2014	 91.9%	 0%	 66.7%	 100%	 0	6.1
Dr.Web Security Space 9	 94.7%	 0%	 33.3%	 50%	 0	5.0

Рисунок 1 – Результаты тестирования

## Литература

1. Лаборатория Касперского [Электронный ресурс]/ ЗАО «Лаборатория Касперского». – Москва, 1997. – Режим доступа: <http://www.kaspersky.ru> – Дата доступа: 08.03.2015.

2. Comss.ru [Электронный ресурс]/ Computer Security Software Russia. – Москва, 2007. – Режим доступа: <http://www.comss.ru> – Дата доступа: 09.03.2015.

3. Securelist – всё об интернет-безопасности [Электронный ресурс] / ЗАО «Лаборатория Касперского». – Москва, 1997. – Режим доступа: <http://www.securelist.ru> – Дата доступа: 10.03.2015.

**П.С. Родинченко (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Леванцов**, ст. преподаватель

## РЕАЛИЗАЦИЯ ПОДСИСТЕМЫ УЧЕТА СЖИЖЕННОГО ГАЗА

На современном этапе развития общества, учет товаров и услуг играет важнейшую роль, являясь одним из главных производственных активов предприятия. На предприятии любого масштаба необходимо вести точный складской учет, оптимально распределять нагрузку между подразделениями и складами.

Эффективный учет – это четко выстроенная система использования ресурсов предприятия. При наличии оптимального количества товаров и услуг такая система позволяет достигать поставленных целей в кратчайшие сроки и с минимальными затратами. Она помогает

увеличивать продуктивность работы, расширять инновационную деятельность, повышать удовлетворенность клиентов, обеспечивать целостность бизнес-процессов.

Основная проблема, с которой сталкиваются сотрудники складов – это большое количество номенклатуры, огромное количество задач, функций, процессов, которыми необходимо оперативно и качественно управлять.

Все вышеперечисленные проблемы практически невозможно решить без применения современных средств сбора и обработки информации. Именно с использованием современных программных комплексов достигаются высокие результаты в скорости получения информации и удобства работы с ней. Весь этот процесс перехода от старых принципов работы к информационным технологиям называется автоматизацией.

Подсистема учета сжиженного газа предполагает создание общей системы, при которой осуществление операций было бы простым и высокопроизводительным. Для этого настраиваются необходимые программы, задействованные в процессе управления сотрудниками.

Возможности автоматизации:

1. Получение руководством предприятия полной аналитической информации, необходимой для принятия решений.
2. Снижение затрат на основные функции для руководителей.
3. Автоматизация работы по учету и ведению документации.
4. Автоматическое формирование необходимых отчетов.

Подсистема учета сжиженного газа решает следующие задачи:

1. Работа с различными типами структур и ведение складского учета.
2. Внедрение технологий для упрощения ведения учета.
3. Ведение и управление информацией о товарах и услугах.
4. Планирование и контроль над деятельностью и взаимодействием между складами, а так же создание сложных планов.
5. Контроль учета оказанных услуг.
6. Анализ эффективности денежных вложений и финансовых затрат.
7. Формирование отчетов для руководства предприятия.

Преимущества автоматизации:

1. Поддержка различных бизнес-процессов для системы учета сжиженного газа.
2. Организация информационного пространства документооборота в подразделениях, исключение дублирования информации.
3. Синхронизация данных между отделами, централизация хранения и обновления информации.
4. Получение отчетности по всему предприятию и упрощение работы по формированию отчетов.

5. Разграничение прав доступа к данным, фиксирование времени и места внесения изменений в информационную базу.

6. Легкость внедрения и освоения работы с системой для любого пользователя.

Таким образом, подсистема учета сжиженного газа является действенным инструментом, облегчающим действия сотрудников предприятия.

### **Литература**

1. Баронов, В.В. Автоматизация управления предприятием / В.В. Баронов. – М.: ИНФРА, 2000. – 239 с.

2. Веснин, В.Р. Управление персоналом. Теория и практика: учебник / В.Р. Веснин. – М.: Проспект, 2008. – 688 с.

3. Дэйв, Ульрих. Эффективное управление персоналом. Новая роль HR-менеджера в организации / Ульрих Дэйв. – М.: Вильямс, 2007. – 304 с.

4. Зорина, Э.О. Критерии эффективности управления персоналом / Э.О. Зорина.

5. Лари, Стаут. Управление персоналом. Настольная книга менеджера / Стаут Лари. – М.: Добрая книга, 2006. – 536 с.

6. Титоренко, Г.А. Информационные системы в экономике: учебник для студентов вузов, обучающихся по экономическим специальностям / Г.А. Титоренко. – М.: ЮНИТИ, 2006.

7. Саак, А.Э. информационные системы управления / А.Э. Саак. – СПб.: Питер, 2008. – 124 с.

**П.С. Родинченко (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Леванцов**, ст. преподаватель

### **АВТОМАТИЗАЦИЯ УЧЕТА СЖИЖЕННОГО ГАЗА ПРЕДПРИЯТИЯ УП «МИНСКГАЗ»**

Одним из весомых условий успешной профессиональной деятельности на современном рынке труда среди экономистов, работников финансовых служб, бухгалтеров и менеджеров является знание и свободное владение навыками работы с системой «1С: Предприятие 8.2». Корпоративная информационная система «1С: Предприятие 8.2» является гибкой настраиваемой системой, с помощью которой можно решать широкий круг задач в сфере автоматизации деятельности предприятий.

Современные программы 1С 8 позволяют в комплексе автоматизировать учет и управление Вашей организации независимо от сферы деятельности, формы собственности и степени сложности.



Любая современная организация, будь то коммерческая фирма или государственное учреждение, стремится к успеху и развитию бизнеса в целом. Очевидно, что сегодня без структуризации и автоматизации ключевых бизнес-процессов эффективная работа немыслима. Внутренний учет, контроль и управление предприятием, взаимодействие с клиентами и партнерами, отношения с финансово-кредитными учреждениями – вся текущая деятельность компаний связана с постоянным использованием автоматизированных информационных систем. В большинстве случаев используются программные продукты 1С 8.

Программное обеспечение 1С решает множество задач учета и управления в организациях различных отраслей и сфер деятельности вне зависимости от типа финансирования компании. Значительного эффекта удастся добиться, когда внедрение сочетается с качественным обучением персонала работе в 1С. На сегодняшний день система прикладных решений фирмы «1С» позволяет осуществить комплексную автоматизацию как производственных, торговых, так и сервисных предприятий. Важная роль при этом в программе 1С 8.2 отводится автоматизации процессов управления финансами холдингов и отдельных компаний, ведению бухгалтерского и налогового учета, расчета заработной платы и управления кадрами, а так же управления складами.

Исходя из отрасли, вида деятельности предприятия, специфики производства или оказываемых услуг задачи учета и управление могут существенно отличаться. Однако единым для всего бизнеса является строгое соответствие текущему законодательству при ведении бухгалтерского и налогового учета.

По мере развития компании, роста производства естественно возрастают и потребности в автоматизации новых задач с целью эффективного использования ресурсов и упрощения трудоемких процессов бизнеса. Благодаря единой технологической платформе программные продукты 1С всегда открыты для доработки, удовлетворяющей как актуальным, так и вновь возникающим потребностям организаций. Высокотехнологичные программы 1С 8 сочетают в себе достоинства массового, заслуживающего доверие пользователей продукта и динамично развивающейся, гибкой, настраиваемой информационной системы, способной реагировать на любое изменение бизнес-структуры.

Помимо качества, 1С 8 отличается широким спектром областей применения, начиная от офисного, заканчивая домашним. Удобная в использовании как для начинающих пользователей, так и для разработчиков программного обеспечения, программа 1С завоевала популярность в России, Украине, Казахстане, Белоруссии, а также успешно применяется в других странах мира. На сегодняшний день наиболее

распространенными среди пользователей являются прикладные решения на базе технологической платформы «1С:Предприятие 8.3».

«1С: Предприятие» – программный продукт компании 1С, предназначенный для автоматизации деятельности на предприятии.

Существует одна платформа и множество конфигураций. Для функционирования какого-либо прикладного решения всегда необходима платформа и какая-либо (одна) конфигурация. Сама по себе платформа не может выполнить никаких задач автоматизации, так как она создана для обеспечения работы какой-либо конфигурации. То же самое с конфигурацией: чтобы выполнить те задачи, для которых она создана, необходимо наличие платформы, управляющей её работой.

Для того чтобы обеспечить работу, система «1С: Предприятие» имеет различные режимы работы: 1С: Предприятие и Конфигуратор.

Режим 1С: Предприятие является основным и служит для работы пользователей системы. В этом режиме пользователи вносят данные, обрабатывают их и получают итоговые результаты.

Режим Конфигуратор позволяет выполнять изменение существующих прикладных решений и создание новых, а также выполнять действия по администрированию информационной базы. Именно этот режим и предоставляет инструменты, необходимые для модификации существующей или создания новой конфигурации.

Конфигуратор предоставляет разработчику все необходимые инструменты для «визуального» описания структуры прикладного решения, создания форм диалогов и выходных документов, механизмов работы пользователей с данными (интерфейсов) и прав доступа различных групп пользователей к различной информации. Кроме этого конфигуратор позволяет создать и настроить взаимодействие различных элементов прикладного решения друг с другом и написать модули на встроенном языке для обработки различных событий и реализации специфических алгоритмов взаимодействия, обработки входных и выходных данных.

На современном этапе развития общества, учет товаров и услуг играет важнейшую роль, являясь одним из главных производственных активов предприятия. На предприятии любого масштаба необходимо вести точный складской учет, оптимально распределять нагрузку между подразделениями и складами.

Эффективный учет – это четко выстроенная система использования ресурсов предприятия. При наличии оптимального количества товаров и услуг такая система позволяет достигать поставленных целей в кратчайшие сроки и с минимальными затратами. Она помогает увеличивать продуктивность работы, расширять инновационную

деятельность, повышать удовлетворенность клиентов, обеспечивать целостность бизнес-процессов.

Основная проблема, с которой сталкиваются сотрудники складов – это большое количество номенклатуры, огромное количество задач, функций, процессов, которыми необходимо оперативно и качественно управлять.

Все вышеперечисленные проблемы практически невозможно решить без применения современных средств сбора и обработки информации. Именно с использованием современных программных комплексов достигаются высокие результаты в скорости получения информации и удобства работы с ней. Весь этот процесс перехода от старых принципов работы к информационным технологиям называется автоматизацией.

Подсистема учета сжиженного газа предполагает создание общей системы, при которой осуществление операций было бы простым и высокопроизводительным. Для этого настраиваются необходимые программы, задействованные в процессе управления сотрудниками.

Возможности автоматизации:

1. Получение руководством предприятия полной аналитической информации, необходимой для принятия решений.

2. Снижение затрат на основные функции для руководителей.

3. Автоматизация работы по учету и ведению документации.

4. Автоматическое формирование необходимых отчетов.

Подсистема учета сжиженного газа решает следующие задачи:

1. Работа с различными типами структур и ведение складского учета.

2. Внедрение технологий для упрощения ведения учета.

3. Ведение и управление информацией о товарах и услугах.

4. Планирование и контроль над деятельностью и взаимодействием между складами, а так же создание сложных планов.

5. Контроль учета оказанных услуг.

6. Анализ эффективности денежных вложений и финансовых затрат.

7. Формирование отчетов для руководства предприятия.

Преимущества автоматизации:

1. Поддержка различных бизнес-процессов для системы учета сжиженного газа.

2. Организация информационного пространства документооборота в подразделениях, исключение дублирования информации.

3. Синхронизация данных между отделами, централизация хранения и обновления информации.

4. Получение отчетности по всему предприятию и упрощение работы по формированию отчетов.

5. Разграничение прав доступа к данным, фиксирование времени и места внесения изменений в информационную базу.

6. Легкость внедрения и освоения работы с системой для любого пользователя.

Таким образом, подсистема учета сжиженного газа является действенным инструментом, облегчающим действия сотрудников предприятия.

### Литература

1. Баронов, В.В. Автоматизация управления предприятием / В.В. Баронов. – М.: ИНФРА, 2000. – 239 с.

2. Веснин, В.Р. Управление персоналом. Теория и практика: учебник / В.Р. Веснин. – М.: Проспект, 2008. – 688 с.

3. Дэйв, Ульрих. Эффективное управление персоналом. Новая роль HR-менеджера в организации / Ульрих Дэйв. – М.: Вильямс, 2007. – 304 с.

4. Зорина, Э.О. Критерии эффективности управления персоналом / Э.О. Зорина.

5. Лари, Стаут. Управление персоналом. Настольная книга менеджера / Стаут Лари. – М.: Добрая книга, 2006. – 536 с.

6. Титоренко, Г.А. Информационные системы в экономике: учебник для студентов вузов, обучающихся по экономическим специальностям / Г.А. Титоренко. – М.: ЮНИТИ, 2006.

7. Саак, А.Э. информационные системы управления / А.Э. Саак. – СПб.: Питер, 2008. – 124 с.

**В.А. Рубин (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Д. Левчук**, канд. техн. наук, доцент

### ВИЗУАЛИЗАЦИЯ ПОИСКОВЫХ ЗАПРОСОВ

Сегодня любое продвижение интернет-ресурса, которое базируется на покупке ссылок, сталкивается с тем, что поисковики придумывают все новые и новые алгоритмы ранжирования и фильтрации, поэтому продвигаться становится все сложнее и сложнее. Приходится работать все с большим количеством пессимизированных сайтов, придумывать, как вывести их из-под санкций поисковых систем, и при всем при этом результат никогда невозможно предсказать, и, конечно же, все это сказывается на получаемом доходе ресурса.

Несмотря на все усилия поисковиков, направленные на борьбу с попытками оптимизаторов влиять на поисковую выдачу с помощью использования внешних ссылок, бэклинки остаются одним из ключевых

факторов, которые учитываются при ранжировании сайтов. Последние тенденции в поисковой оптимизации подтверждают смещение в сторону качества, а не количества, поскольку ссылки, которые, по мнению поисковой системы, не являются достаточно качественными или «спамными», просто теряют «вес», и становятся абсолютно бесполезными. Больше того, за попытки манипулирования поисковой выдачей на штраф могут быть наложены штрафные санкции, что может отразиться на позициях и усложнит продвижение.

Существует огромное количество ресурсов, которые помогают более эффективно использовать поисковую оптимизацию (search engine optimization, далее по тексту SEO). Например: Ahrefs (<http://ahrefs.com>), Plagspotter, SemRush, WhiteSpark. Каждый из указанных сервисов позволяет вести учет метрик ресурса и его отдельных страниц по различным параметрам.

К примеру, сервис Ahrefs является инструментом для анализа бэклинков, ведущих на сайт, с помощью которого можно анализировать ссылочную массу сайтов-конкурентов, и на основе полученных данных вырабатывать или совершенствовать свою стратегию наращивания внешних ссылок.

Защита авторских прав и уникальность контента являются одними из наиболее острых проблем, которые затрагивают множество пользователей интернета. На сегодняшний день разработано множество различного программного обеспечения и веб-сервисов, которые предоставляют возможность решать такие проблемы. Одним из таких веб-сервисов стал ресурс от компании Plagspotter. Изначально, данный функционал разрабатывался для личного использования внутри компании, но после проведения сложной работы, было принято решение представить данное «изобретение» каждому желающему. Благодаря веб приложению Plagspotter, каждый желающий может проверить тексты сайта, указывая адрес страницы или же провести полную проверку сайта. После ввода необходимого для проверки адреса, Plagspotter начинает выделять текст со страниц, разбивает его на несколько частей и проверяет на плагиаты, используя сразу три поисковых системы, это Google, Bing и Yahoo. Так же хотелось бы отметить, что проверяемые тексты проходят дополнительную проверку на опечатки и перестановку слов, благодаря чему пользователи добиваются уникальности текстов.

WhiteSpark предоставляет инструментарий для локального SEO-исследования бизнес-рынка. С помощью сервиса Whitespark можно найти упоминания вашего бизнеса локально(в рамках города или страны), с удобной визуализацией полученных данных, а так же четко отслеживать свой рейтинг в любом городе через карты.

SemRush – это сервис анализа ключевых слов, который имеет интуитивный интерфейс. Вводите домен, кликаете на кнопку и получаете данные позиций, анализ качества ключевых запросов, некоторые данные по конкурентам и многое-многое другое. Работает этот сервис пока только с Google. С помощью SemRush можно быстро собрать список ключевых слов из Google, по которым ваш сайт сейчас находится в топе с точным указанием страниц. Также в итоговом отчете можно увидеть частотность запроса по Google и среднюю стоимость клика в AdWords по указанному запросу. Есть такие показатели как конкуренция запроса, количество сайтов в выдаче по этому запросу, общее число трафика запроса по отношению ко всему трафику с Google на сайт.

Также можно посмотреть конкурентов сайта. А после оценить каждого и посмотреть запросы, по которым он продвигается. Также можно увидеть общие запросы с конкурентами данного сайта. Для этого надо кликнуть по вкладке «Конкуренты в поисковой выдаче» и далее выбрать домен, с которым необходимо провести сравнение.

В результате, используя вышеуказанные сервисы и предоставляемый ими интерфейс программирования приложений, можно реализовать веб-сервис, который позволит анализировать всевозможные SEO-метрики сайта в едином месте, без необходимости отслеживания отдельных метрик по отдельным ресурсам.

Для реализации такого проекта были выбраны следующие инструменты:

- язык программирования Python,
- веб-фреймворк Django,
- база данных PostgreSQL,
- JavaScript-фреймворк AngularJS,
- JavaScript-библиотека визуализации данных HighCharts,
- HTML-шаблонизатор Bootstrap.

Проектирование веб-интерфейса проводилось при помощи библиотеки Twitter Bootstrap, что позволило разработать простейший интерфейс приложения в сжатые сроки.

Для упрощения части приложения, ведущей диалог с клиентом на стороне браузера, был использован AngularJS, что позволило упростить организацию кода на стороне клиента (браузер).

Взаимодействие пользователя с сервисом можно описать так:

1. Клиент выбирает сервис, по которому он желает получить метрики.

2. Клиент вводит необходимую информацию для запроса к сервису (адрес сайта, адрес страницы сайта, домен ресурса, текст, и т. д.)

3. Запрос клиента записывает в базу данных информацию о нем (сервис, запрос, дата запроса). Так же данный запрос попадает в очередь запросов клиентов

4. Все запросы клиентов поочередно достаются из базы данных, после чего отправляются на сторонний ресурс и после прихода ответа от них – обратно записываются в базу данных к нужному запросу.

5. Клиенту в удобном визуализированном диаграммами виде показывается информация по его запросу, на основании которой он может провести SEO-аудит своего ресурса и выдвинуть дальнейшую стратегию продвижения.

**В.А. Рубин (УО «ГГУ им Ф. Скорины», Гомель)**

Науч. рук. **В.Д. Левчук**, канд. техн. наук, доцент

## **АСИНХРОННЫЕ ЗАДАНИЯ В DJANGO ДЛЯ ВЗАИМОДЕЙСТВИЯ С API СТОРОННИХ СЕРВИСОВ**

В проекте, где необходимо вести постоянный обмен данными со сторонними сервисами посредством предоставляемого ими API, необходимо заранее подготовиться к серьезным нагрузкам на серверную часть.

В условиях высоких нагрузок недопустимо уже будет использовать синхронную модель взаимодействия, где ваш сервис произведет вызов на сторонний адрес, подождет от него ответ и потом произведет какие-либо операции по получению ответа. Ведь этого ответа можно не дожидаться (недоступен ресурс), а возможно сервер просто «ляжет» под такими нагрузками.

Именно для таких нужд применяются асинхронные задания. Особенно красиво они реализованы в Python веб-фреймворке Django, а именно в одной из вспомогательных библиотек – Celery.

Celery – «distributed task queue». Это распределенная асинхронная очередь заданий, которая позволяет:

- выполнять задания асинхронно или синхронно, распределенно, одновременно,
- выполнять периодические и отложенные задания,
- ограничивать количество заданий в единицу времени (rate limit, для задания или глобально),
- мониторить выполнение заданий,
- присылать отчеты об ошибках на email,
- проверять, выполнилось ли задание (удобно для построения Ajax приложений, где клиент ждет факта завершения).

Для использования Celery в Django-проекте необходимо использовать сервис очередей. Ввиду особенностей проекта был выбран сервис RabbitMQ.

Такая организация проекта позволяет ему не «упасть» под высокой нагрузкой и не даст данным пользователя потеряться в случае неуспешного ответа со стороны сервиса.

Детали использования библиотеки Celery в производственном проекте обсуждаются в докладе.

**А.Е. Руденко (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **П.Л. Чечет**, канд. техн. наук

## **ФРЕЙМВОРК CORE DATA ДЛЯ ОПЕРАЦИОННОЙ СИСТЕМЫ IOS**

Используя компьютеры для выполнения своих задач, люди рассчитывают, что внесенные ими изменения будут сохранены. Сохранение изменений играет важную роль в офисных программных пакетах, текстовых редакторах, играх, браузерах и т. д. Большинство программного обеспечения нуждается в возможности хранить введенные пользователем данные для последующего восстановления состояния работы, но конечно же есть и такое ПО, которое в этом не нуждается – калькуляторы, новостные ленты, будильники, виджеты о погоде. Понимание того, каким образом можно хранить данные на устройствах, является критически важным при разработке продвинутых приложений.

Core Data – это фреймворк для манипуляции с данными у графов объектов, предназначенный для операционных систем Mac OS X и iOS. Он был представлен для Mac OS X, начиная с версии 10.4 Tiger и для iOS, начиная с iPhone SDK 3.0. Фреймворк организует данные с помощью реляционной модели, позволяющей сериализовать атрибуты сущностей в формат XML, бинарную форму или сохранять их в таблицы SQLite. Core Data является фреймворком для работы с данными, которая позволяет работать с сущностями и их связями (отношениями к другим объектами), атрибутами, в том виде, который напоминает работы с объектным графом в обычном объектно-ориентированном программировании. Т. е. данными можно манипулировать с помощью высокоуровневых объектов языка (например, Objective-C), которые представляют собой сущности предметной области и связи между ними.



Core Data не проектировался с целью заменить все возможные варианты хранения данных, которые могут лучше подойти при решении определенных задач (например, хранение неструктурированной информации в файлах или хранение пар значений вида ключ-значение в специализированных СУБД), однако он тесно интегрирован с другими фреймворками от Apple, например с Cocoa Touch. Core Data скрывает большинство деталей по работе с хранилищем данных, позволяя разработчику сконцентрироваться на описании бизнес-логики приложения.

**А.Е. Руденко (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **П.Л. Чечет**, канд. техн. наук

## **ИСПОЛЬЗОВАНИЕ ШАБЛОНА ПРОЕКТИРОВАНИЯ «ПУЛ ОБЪЕКТОВ» В ИГРОВОМ ПРИЛОЖЕНИИ**

Объектный пул (англ. object pool) – порождающий шаблон проектирования, набор инициализированных и готовых к использованию объектов. Когда системе требуется объект, он не создаётся, а берётся из пула. Когда объект больше не нужен, он не уничтожается, а возвращается в пул.

Шаблон проектирования используется для улучшения производительности и эффективности использования памяти за счет повторного использования объектов из фиксированного пула, вместо их индивидуального выделения и освобождения.

Программирование для игровых консолей или мобильных телефонов больше похоже на программирование для встроенных систем, чем на программирование для РС. Как и в программировании для встроенных систем, консольные или мобильные игры должны работать очень долгое время без падений и утечек памяти, при том что эффективные менеджеры памяти встречаются не так уж и часто. В такой рабочей среде фрагментация памяти смертельно опасна.

Даже если фрагментация встречается нечасто, она может постепенно привести кучу (heap) в состояние бесполезных пузырей из дырок и щелей, полностью лишив игру возможности с ней работать.

Большинство консольных платформодержателей требуют, чтобы игры проходили «тест на протечку» (soak test), когда игра оставляется работающей на несколько дней в демо-режиме. Если игра падает – ей не разрешают выйти. Тест на протечку иногда проваливается и из-за какой-нибудь редкой ошибки, но чаще всего игру обрушивает утечка памяти, вызванная фрагментацией.

Пул объектов обладает следующими преимуществами: с точки зрения менеджера памяти он представляет собой единую непрерывную область в памяти, освобождаемую только после завершения игры; с точки зрения программиста (пользователя пула), он предоставляет простую возможность создавать и удалять объекты.

Этот шаблон широко используется в играх не только для очевидных вещей типа игровых сущностей и визуальных эффектов, но и для менее заметных структур данных типа проигрываемых звуков.

Пул объектов можно использовать, когда программе нужно часто создавать и удалять однотипные объекты, либо, если выделение объектов из кучи работает медленно или может привести к фрагментации памяти, или, например, когда каждый объект инкапсулирует ресурс типа базы данных или сетевого соединения, который сложно получить и можно использовать повторно.

Размер пула необходимо настраивать соразмерно с нуждами игры. При настройке обычно проще всего понять когда пул недостаточного размера. В тоже время, необходимо следить и за тем чтобы пул не был слишком большим. Если уменьшить пул, освободившуюся память можно использовать для чего-либо более полезного.

Одним из ограничений пула является то, что в каждый момент времени может быть активно только определенное (заранее известное) количество объектов. Разделение памяти на отдельные пулы для различных типов объектов означает с одной стороны то что последовательность взрывов не заставит, к примеру, систему частиц использовать всю доступную память, не позволив программисту создать на игровой сцене что-либо более полезное, вроде нового противника.

Кроме того, программист должен быть готов к ситуации, когда выделить объект из пула не удастся, потому что все объекты заняты. Есть несколько стратегий обработки такой ситуации.

Прямое вмешательство. Это самое очевидное «исправление»: будем настраивать размер пула таким образом, чтобы он никогда не переполнялся независимо от действий пользователя. Для пулов с важными объектами, такими как противники или геймплейные предметы, это хороший выход. Не может быть «правильной» обработка недостатка свободных слотов для создания большого босса, когда игрок дошел до конца уровня. Недостатком этого метода является то, что программисту придется держать занятыми большие объемы памяти ради каких-то редких крайних случаев. Поэтому фиксированный размер пула не может считаться лучшим решением для всех состояний игры. Например, некоторые игровые уровни могут требовать расширенных пулов для визуальных эффектов, а другие уровни – для звуков.

В таких случаях лучше иметь пулы объектов, настраиваемые отдельно для каждого из сценариев.

Игнорирование создания объекта. Тип решения, который имеет смысл для некритичных частей игры, например, при работе с системами частиц. Пользователь скорее всего не обратит внимание если следующий взрыв будет немного менее впечатляющим, чем уже отображаемые на экране.

Принудительное уничтожение существующего объекта. Для примера можно использовать полностью заполненный пул проигрываемых в данный момент в игре звуков. При этом планируется, что игнорирование создания нового звукового эффекта будет негативно воспринято пользователем, который уже привык слышать подобные звуки на определенные действия (например, на действие перезарядки оружия). В данном случае есть смысл найти самый тихий звук из тех, что уже играют и заменить его новым звуком. Новый звук заглушит слышимый обрыв предыдущего звука. В целом, если исчезновение существующего объекта будет менее заметным чем неоявление нового – это вполне хорошее решение.

Увеличение размера пула. В случае, если менеджер памяти позволяет программисту распоряжаться памятью более гибко, он может увеличивать размер пула во время выполнения или создавать дополнительные пулы переполнения. Однако динамические пулы страдают от всех недостатков обычного управления динамическими структурами данных и могут быть неприменимы в случае с ограничениями на производительность игровой платформы.

Большинство менеджеров памяти обладают отладочными функциями, которые очищают только что выделенную или освобожденную память определенными значениями, к примеру **0xdeadbeef**. Это помогает обнаруживать болезненные ошибки, вызванные использованием неинициализированных значений или обращением к уже освобожденной менеджером памяти.

Так как наш пул объектов не заходит в деле управления памяти дальше повторного использования объектов, он не имеет подобной «страховочной сетки». Еще хуже то, что память, используемая для нового объекта хранила раньше объект того же самого типа. Это делает весьма возможной ситуацию, когда программист забудет инициализировать что-то внутри нового созданного объекта, а память, где он будет размещаться, уже будет содержать почти корректные данные, оставшиеся от прошлого объекта. Необходимо с особой тщательностью следить за тем, чтобы код инициализации нового объекта в пуле выполнял инициализацию объекта полностью.

Пулы объектов реже всего используются в системах со сборщиками мусора, потому что в таком случае менеджер памяти сам занимается проблемой фрагментации. Но пулы все равно полезны тем, что помогают программисту избегать выделения и освобождения памяти. Особенно актуально подобное выделение памяти на мобильных устройствах с медленным процессором и простым сборщиком мусора.

В этом случае следует опасаться потенциальных конфликтов. Так как пул на самом деле не освобождает объекты когда они больше не используются, они остаются в памяти. Если они содержат ссылки на другие объекты, они тем самым не дадут сборщику утилизировать и эти объекты тоже. Чтобы этого избежать, нужно очищать все ссылки на другие объекты, когда объект из пула нам больше не нужен.

**Е.А. Рушнов (БГУ, Минск)**

Науч. рук. **И.М. Гулис**, д-р физ.-мат. наук, профессор

## **ОПТИЧЕСКАЯ СИСТЕМА ДИСПЕРСИОННОГО МНОГОЦЕЛЕВОГО СПЕКТРОМЕТРА**

Создание приборов, позволяющих получать спектроскопическую информацию об объектах с пространственным разрешением, рассматривается в настоящее время как одно из магистральных направлений в современной спектроскопии.

Широко используемые в настоящее время в СПР аппаратурно-методические решения сводятся в основном к различным вариациям двух подходов: мультizonальной (мультиспектральной) съемке (используются светофильтры) и гиперспектрологии, заключающейся в последовательной съемке спектров участков изображения объекта. В обоих случаях спектрально-пространственная информация об объекте может быть сформирована в т. н. гиперкуб  $I(x, y, \lambda)$  – зависимость интенсивности от двух пространственных и спектральной координат.

Целью настоящей работы является отработка решений по ключевым элементам (объективы, дисперсионное устройство) дисперсионного многоцелевого спектрометра (ДМС) для видимой области спектра (400–800 нм), а также построение оптической схемы и ее оптимизация. Проектируемый ДМС должен позволять зарегистрировать гиперкуб с размерностью до  $N_x \cdot N_y \cdot N_\lambda = 400 \cdot 300 \cdot 40$  в одном акте измерения. Оценки показывают, что для реализации требования по спектральному разрешению (порядка 10 нм) диаметр кружка рассеяния, даваемого коллиматорным и камерным объективами, должен не превышать 10 мкм.

Таким образом, оба объектива должны обеспечивать разрешение на уровне 50 линий/мм. Среди доступных объективов этим требованиям удовлетворяет Nikkor 50mm 1/1.8D.

Другим важным комплектующим спектрометра является дисперсионный элемент. Использование в качестве дисперсионного элемента дифракционной решетки невозможно вследствие наличия множества порядков дифракции, поэтому приходится остановиться на призме. Однако призмы имеют сильную зависимость угловой дисперсии от длины волны (различие может превышать порядок для диапазона 400–800 нм), которая минимизируется использованием составных призм [1, 2] (рисунок 1). Применение таких призм позволяет, путем подбора на стадии расчетов марок стекол и преломляющих углов каждой призмы, изменять как угол отклонения, так и дисперсию.

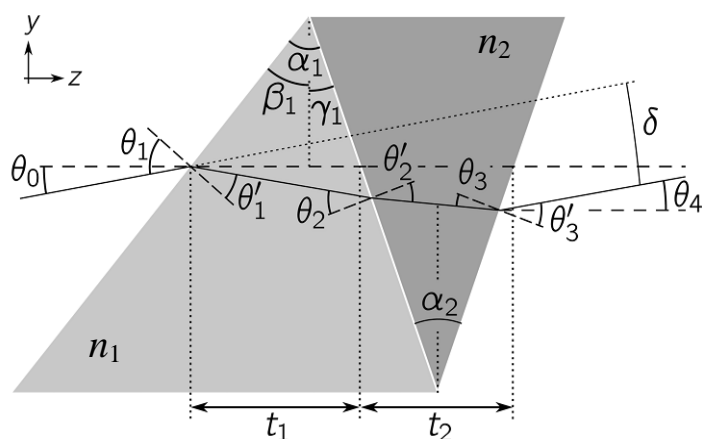


Рисунок 1 – Составная призма

Для моделирования составного дисперсионного элемента была выбрана призма, состоящие из стекол ТФ5 и ЛК7. Расчеты проводились для угла наименьшего отклонения первого элемента (ТФ5). Угловая дисперсия для составной призмы определяется по следующей формуле

$$\frac{d\delta}{d\lambda} = \frac{\sin(\alpha_1 / 2)}{\sqrt{1 - n_1^2 \sin^2(\alpha_1 / 2)}} \frac{dn_1}{d\lambda} + \frac{1}{\sqrt{1 - n_2^2 \sin^2(\alpha_2 - \arcsin(\frac{n_1}{n_2} \sin(\alpha_1 / 2)))}} \times$$

$$\times (\sin(\alpha_2 - \arcsin(\frac{n_1}{n_2} \sin(\alpha_1 / 2))) \frac{dn_2}{d\lambda} + n_2 \cos(\alpha_2 - \arcsin(\frac{n_1}{n_2} \sin(\alpha_1 / 2))) \times$$

$$\times (-\frac{1}{\sqrt{1 - (n_1 / n_2)^2 \sin^2(\alpha_1 / 2)}} (\frac{\sin(\alpha_1 / 2)}{n_2^2} (n_2 \frac{dn_1}{d\lambda} - n_1 \frac{dn_2}{d\lambda}))).$$

Задаваясь требованием длины каждого из миниспектров в 300 мкм, что отвечает разности углов отклонения  $0,34^\circ$ , а также требованием минимального изменения угловой дисперсии, после проведения расчетов и оптимизации, получены преломляющие углы  $10^\circ$  и  $42^\circ$  для ТФ5 и ЛК7 соответственно.

Для составной призмы значение угловой дисперсии на синем краю, больше чем на красном в 3,6, что в 3 раза меньше чем у одиночных призм. На рисунке 2 показана зависимость угловой дисперсии от длины волны для составной призмы и отдельно для каждого элемента при условии, что угол падения излучения на призму равен углу наименьшего отклонения для первого элемента (стекло ТФ5).

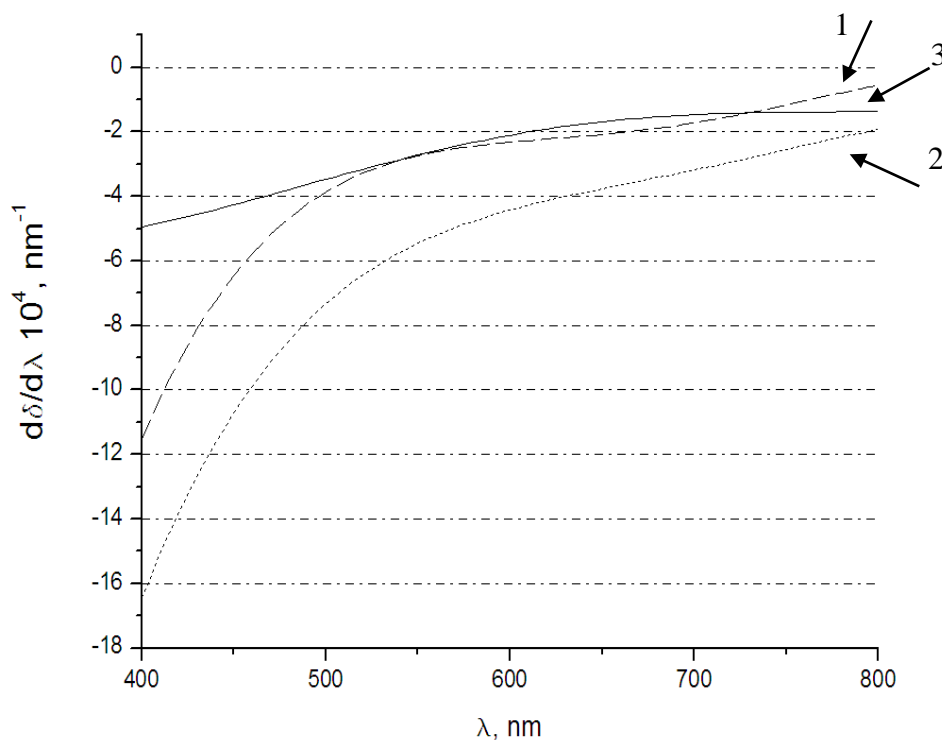


Рисунок 2 – Зависимость угловой дисперсии от длины волны для:  
1 – ТФ5, 2 – ЛК7, 3 – составная призма

Используя полученные результаты для преломляющих углов элементов составной призмы, построена оптическая схема дисперсионного спектрометра и проведена ее оптимизация в оптическом дизайнера Zemax (рисунок 3). Средний угол отклонения луча от первоначального распространения согласно аналитическим расчетам составляет  $19,4^\circ$ , а по данным из оптического дизайнера  $19,2^\circ$ . Таким образом, полученные результаты вполне удовлетворительны и обеспечивают выполнение требований предъявляемых к дисперсионному многощелевому спектрометру, а также позволяют приступить к изготовлению опытного образца спектрометра.

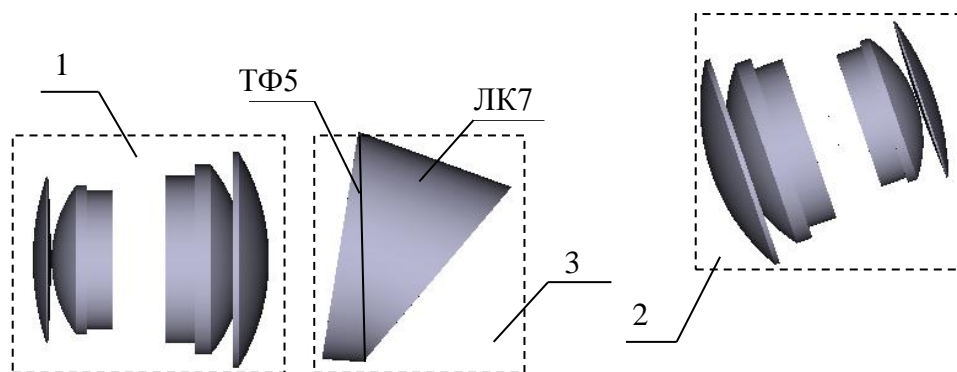


Рисунок 3 – Оптическая схема дисперсионного спектрометра, где 1 – коллиматорный объектив, 2 – камерный объектив, 3 – составная призма

### Литература

1. Hagen N., Tkaczyk T. Compound prism design principles, I // Appl Opt. 2011. V. 50. N. 25. – P. 4998–5011.
2. Brodzik AK, Mooney JM. Convex projections algorithm for restoration of limited-angle chromotomographic images // J. Opt. Soc. Am. A. 1999. – P. 246–257.

**А.В. Рыбакова** (УО «ГГТУ имени П.О. Сухого», Гомель)  
 Науч. рук. **В.С. Мурашко**, ст. преподаватель

### АВТОМАТИЗАЦИЯ ФОРМИРОВАНИЯ КОНСТРУКТОРСКОЙ ДОКУМЕНТАЦИИ НА ДЕТАЛИ ТИПА «ВТУЛКА» В AUTOCAD

Автоматизация инженерного труда на основе широкого и эффективного использования ЭВМ является одним из элементов комплексной автоматизации современного производства.

Применение компьютерных графических систем для разработки технической документации дает конструктору практически неограниченные возможности: разрабатывать чертежи-аналоги по чертежам-прототипам; создавать библиотеки изображений стандартных элементов (изображения крепежных элементов – болтов, шпилек, гаек и т. п.); моделировать трехмерные геометрические объекты с помощью объемных примитивов и операций, выполняемых с ними; адаптировать графическую систему к решаемым задачам пользователя путем расширения графической системы разработкой собственного меню пользователя и внедрения в систему языков программирования высокого уровня.

Чтобы создать набор формирования чертежей определенного класса деталей, сначала нужно выбрать из уже имеющихся рисунков наиболее сложные и полно отражающие все особенности данного класса. Далее на их основе разрабатывается чертеж типовой детали. Все его размеры должны быть выражены в параметрах. До начала разработки программного обеспечения необходимо выделить в этом чертеже основу детали и функциональные элементы.

Основа любой детали – это заготовка, из которой с помощью последующей обработки (сверления, точения, фрезерования и пр.) получается требуемое изделие. В принципе все основы можно представить как заготовку в форме либо цилиндра, либо параллелепипеда без отверстий, однако на практике заготовки бывают более сложными по форме и в некоторых случаях – со сквозными отверстиями.

Функциональный элемент, с точки зрения разработчика программного обеспечения – это одна параметрическая обработка заготовки. При обработке модели заготовки необходимо корректно модифицировать весь ее чертеж.

Целью данной работы является автоматизация формирования конструкторской документации на детали типа «Втулка» в AutoCAD, которая предусматривает следующую постановку задачи.

Начертить рабочий параметрический эскиз «Типовые втулки».

Проанализировать и выявить все размеры, которые необходимы для программирования чертежей данных типов втулок (выбрано четыре типа).

Продумать последовательность рисования деталей и для каждого типа втулки выявить функциональные элементы.

Написать программу. Размеры деталей должны вводиться в программу пользователем с помощью диалоговых окон, написанных на языке DCL.

Предусмотреть возможность создания диалогового окна выбора формата листа A1, A2, A3, A4 с атрибутами, на котором будет отображаться деталь.

Предусмотреть возможность функций переноса детали в любую указанную пользователем точку.

Запрос пользователю: повторить построение.

Для выполнения поставленной задачи использовалась конкретная реализация языка LISP – встроенный в САПР AutoCAD интерпретатор языка AutoLISP. Выбор этого языка в качестве встроенного для САПР AutoCAD вызван тем, что список – оптимальный способ представления графической информации, а также легкостью реализации и небольшими размерами интерпретатора. Если при генерации AutoCAD



интерпретатор AutoLISPa был подключен, то он загружается в оперативную память после запуска графического редактора AutoCAD и доступен в течение всего сеанса работы с AutoCAD.

Система автоматизированного проектирования AutoCAD предусматривает возможность самостоятельного написания диалоговых окон, отличных от определенных в системе. Для этой цели был разработан специальный язык – DCL (Dialogue Control Language, или другими словами – язык управления диалоговыми окнами).

В результате была разработана программа «Vtulki», которая выполняет следующие функции.

Загрузка диалога DCL.DCL для выбора типа vtulki (рисунок 1).

Загрузка нового диалога для выбранного типа втулки DCL2.DCL (рисунок 2).

Вызов функции ok\_tab, формирующей список данных, являющихся результатом диалога DCL2.

Вызов функции forma для выбора рамки формата.

Рисование выбранной втулки – функция vtulka.

Вызов функции gazmv для расстановки размеров втулки. Функция gazmv вызывает функцию setdim, которая устанавливает необходимые размерные системные переменные.

По выбору пользователя можно переместить по полю чертежа деталь.

Повторить все действия с п. 1 Д/Н?

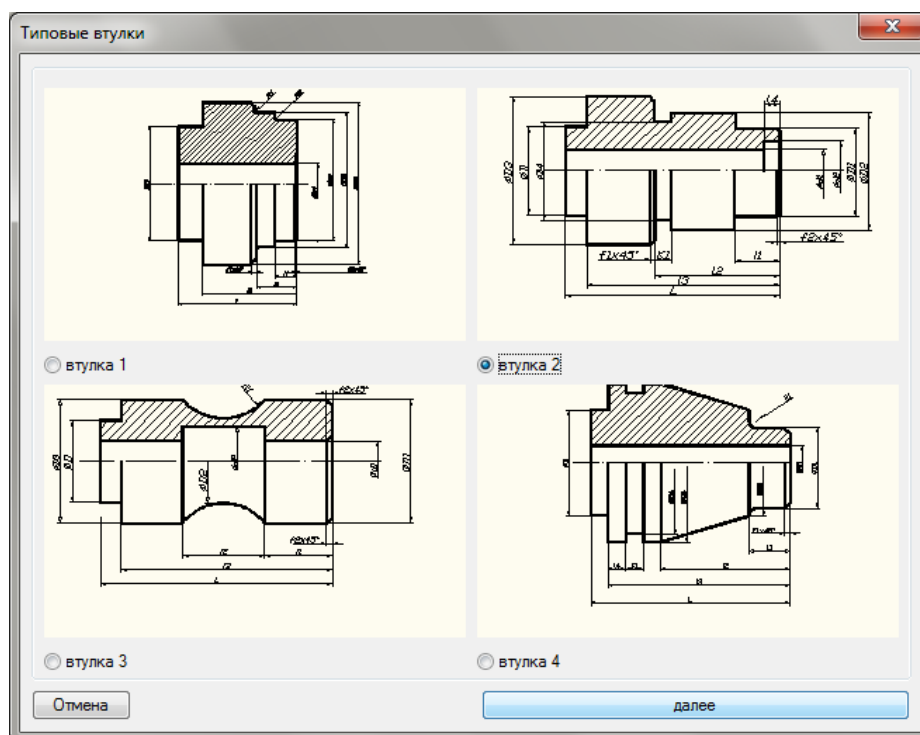


Рисунок 1 – Диалоговое окно выбора вида втулки

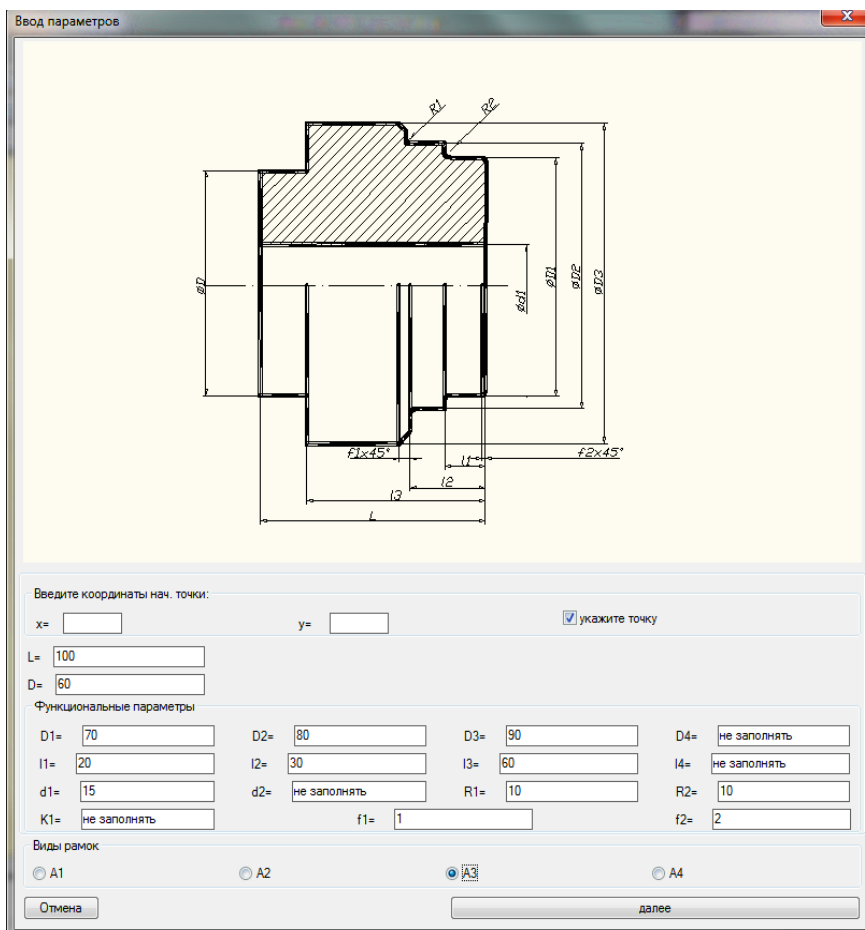


Рисунок 2 – Ввод параметров для выбранной втулки

Программа удобна в обращении и позволяет быстро построить втулку с указанным форматом рамки. Использовать программу для автоматизации формирования конструкторской документации на детали типа «Втулка» могут студенты в курсовых и дипломных работах, а также пользователи – конструкторы.

**Ф.С. Рябиков (УО «ГГУ им. Ф. Скорины», Гомель)**  
 Науч. рук. **В.А. Дробышевский**, ст. преподаватель

## ОСНОВНЫЕ ПРАВИЛА СЪЕМКИ ФОТОГРАФИЙ ДЛЯ СОЗДАНИЯ 3D-ПАНОРАМ

Начнем с представления формата панорамы. Зачастую даже самый широкоугольный объектив не дает нужного угла охвата во время съемки пейзажей или городов. Особенно это заметно на фотоаппаратах с кропом (обрезанием краев изображений). Именно для этих случаев и предназначена возможность объединения нескольких фотографий

в одно изображение, при этом так же резко возрастает качество детализация изображения. Далее будут приведены основные правила, которые следует соблюдать при съемке панорам.

Итак, во время съемки обязательно наложение кадров, которое необходимо при сборке панорам. Вряд ли можно составить панораму из кадров, сделанных встык. По крайней мере, программы этого не умеют. Поэтому все фотографии должны накладываться друг на друга. Лучше если для наложения будут отведены треть кадров, или даже половина. Тогда даже при бракованном кадре будет возможность восстановить его часть из других и панорама не пострадает.

Еще одним правилом при создании панорамы является то, что съемку необходимо вести слева направо. Прежде всего, такой порядок обусловлен удобством последующего просмотра и алгоритмами работы программ для склеивания панорам.

Очень важно фотографировать панораму быстро. Поэтому предварительно стоит оценить предстоящую панораму, затем сделать все необходимые настройки. После завершения подготовительного этапа нужно просто смотреть в видоискатель и делать снимки не прерываясь. Если сделать перерыв в серии кадров, то может, например, измениться освещение и панорама уже не сойдется без проблем. Но и излишняя торопливость тоже не нужна, после перемещения фотоаппарата его нужно четко зафиксировать, особенно при относительно длинных выдержках, чтобы избежать смазывания картинки. В среднем на каждый снимок должно уходить не более 1–2 секунд, т. е. съемка панорамы из 5–7 снимков займет где-то 10 секунд.

**Ф.С. Рябиков (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.А. Дробышевский**, ст. преподаватель

## **ПОСЛЕДОВАТЕЛЬНОСТЬ ДЕЙСТВИЙ ПРИ СОЗДАНИИ 3D-ПАНОРАМЫ**

Начнем с определения сферической панорамы. Сферическая панорама – это фотография с охватом  $360^\circ$  по горизонтали и  $180^\circ$  по вертикали, то есть ваше окружение. При просмотре таких панорам создается эффект присутствия.

Для создания сферической проекции для 3D-панорамы необходимо отснять все окружающее пространство, и «сшить» в специальном программном обеспечении. Кубическая проекция получается путем преобразования сферической.

Панорамы применяются для демонстрации и продвижения товаров и услуг в сферах недвижимости, развлечений, туризма, автобизнеса, торговых площадей, гостиничного бизнеса, спорт-индустрии и других. Их снимают в путешествиях и на различных мероприятиях.

При создании панорамы руководствуются следующими действиями: фотографирование по определенным правилам, и последующая склейка полученных снимков в специализированных программах в единую панораму.

Для создания сферической панорамы нужен штатив, фотокамера, панорамная головка и компьютер. Выбор фотокамеры для съемки панорам, как ни странно, является наименее значимым фактором для качества получаемой картинки. Даже зеркальные камеры любительского ценового диапазона, например, Canon 350D или Nikon D50, удовлетворяют минимальным требованиям. В панорамной съемке недостаточный размер матрицы, независимо от пикселей, компенсируется большим числом кадров и возрастающим временем последующей обработки.

В качестве объективов, применяемых для изготовления виртуальных сферических панорам, обычно используются сверхширокоугольные объективы-фиксы, широкоугольные зум-объективы на коротком конце, либо объективы типа fisheye (фишай, «рыбий глаз»).

Весь процесс состоит из следующих действий: съемка, склейка отдельных кадров в целое панно, ретушь и компиляция во Flash. При создании виртуальных туров используется специализированное программное обеспечение.

При съемке виртуальных 3D-туров в фокусе должно быть все: и передний и средний и задний план. Для этого просто на своем фотоаппарате поставьте значение на бесконечность. Профессиональные фотографы, чтобы добиться наилучшей резкости на снимках, настраивают гиперфокальное расстояние для объектива.

Также в ручном режиме настраивается экспозиция. Выставляются одинаковые значения выдержки и диафрагмы для всех снимков одной панорамы. Это нужно чтобы была возможность склеить отдельные кадры воедино.

Необходимо убедиться, чтобы в помещении не было больших перепадов темных и светлых участков. Для этого можно, например, зашторить окна, а так же подсветить темные участки, используя доступные источники освещения.

При повороте камеры происходит смещение объектов ближнего и дальнего плана относительно друг друга. Чтобы этого избежать, необходимо вращать фотоаппарат вокруг специальной точки, называемой нодальной. Эта точка находится на оптической оси объектива в месте

пересечения лучей. При вращении камеры вокруг нодальной точки смещения объектов ближнего и дальнего плана не происходит. Стоит отметить, что расположение этой нодальной точки индивидуально для каждого объектива. Для того чтобы фотоаппарат можно было вращать вокруг именно этой точки, применяются панорамные головки, которые накручиваются непосредственно на штатив.

При наличии зеркального фотоаппарата, можно определить нодальную точку объектива путем ввода в поисковике запроса вида «модель объектива nodal point».

Теперь начинаем снимать кадр за кадром с перекрытием порядка 20–30 %. Благодаря анализу именно этих областей программа позже сможет сшить все фотографии в единую панораму.

Во время съёмки панорамы, камерой со штатива делают ряд последовательных снимков вокруг своей оси, затем снимают зенит (верх) и надир (низ) – самая сложная часть съёмки.

ISO при съёмке ставится на минимальное значение, если не производится съёмка со штатива и отсутствуют быстро движущиеся объекты.

При съёмке панорамы некоторые части её могут быть по-разному освещены и отличаться по уровням цвета и света друг от друга. Чтобы обеспечить натуральную цветопередачу, перед склейкой исходные кадры обычно подвергаются серьёзной обработке. В сравнении с фотографиями, снятыми в формате RAW, возможности обработки JPEG изображений весьма ограничены. В связи с этим при фотосъёмке рекомендуется устанавливать формат RAW+JPEG. Использование формата RAW производится для минимизации дальнейших проблем. На этапе RAW-конвертора выполняется компенсация грубых перепадов по яркости отдельных фотографий будущей панорамы. А благодаря JPEG можно быстро просматривать снятые изображения.

Конечно же, без панорамной головки съёмка панорамы усложняется. Основное затруднение – обеспечить поворот фотоаппарата вокруг нодальной точки. Помочь в этом деле может опора фотоаппарата на каком-то вертикальном предмете. Это может быть штатив, или на крайний случай, сук или палка.

В таких случаях нужно постараться выбрать сюжет для съёмок, когда ближний план отсутствует, а почти все объекты находятся на дальнем плане. Например, площадь города. Тогда эффект параллакса почти не проявит себя, и панорама нормально «сошьётся».

Безусловно, всем известно, зенит – это линия перпендикуляра вверх от плоскости горизонта в точке, в которой вы находитесь, а надир – вниз. Т. е. съёмка зенита – это съёмка фотоаппаратом, повернутым вертикально вверх, а надира – вертикально вниз.

Съемку надира возможно производить с рук. Главное – обеспечить положение фотоаппарата таким же, как если бы он находился на панорамной головке. Так, сперва он устанавливается на головке вертикально вниз, затем отсоединяется и удерживается в таком положении на вытянутой руке. В это время нужно другой рукой убрать штатив и, в итоге, снять надир.

Съемка зенита – более простая задача. Следует повернуть камеру вертикально вверх и сделать фотографию. В случае если вы снимаете небо, то усложняете работу программе для сшивки фото в панораму по той причине, что ей не за что будет «зацепиться». В таком случае зенит 3D-панорамы придется дорабатывать в графическом редакторе.

**И.Г. Савченко (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **Е.А. Левчук**, канд. техн. наук, доцент

## **РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ ОПТОВО-РОЗНИЧНОГО ЗВЕНА ТОРГОВОЙ ОРГАНИЗАЦИИ**

Для сферы торговли существует много различных программных решений. Для автоматизируемого объекта в качестве платформы была выбрана среда 1С 8.2 Розница для Беларуси в связке с конфигурацией Управление торговлей 3.0.

Каждая часть комплекса обеспечивает охват и управление в своей области. К примеру, 1С 8.2 Розница для Беларуси устанавливается непосредственно в торговой точке и обеспечивает учет и контроль продаж за наличный и безналичный расчет, продаж с учетом акций и скидок, отгрузку товаров по накладным.

В свою очередь Управление торговлей ред. 3.0, взаимодействуя с 1С 8.2 Розница для Беларуси, позволяет анализировать динамику продаж как в целом по организации, так и в разрезе магазинов, товаров, поставщиков и их комбинаций. Так же программа позволяет вести учет приходных и расходных операций по каждому месту хранения (магазину).

В процессе реализации проекта были выполнены следующие доработки программного продукта согласно техническому заданию заказчика:

- автоматизация оприходывания товаров от поставщика по спецификации. Позволяет автоматически загружать приходные накладные от поставщика в соответствии с договором поставки;
- автоматическая рассылка отчетов поставщикам;

– календарь инвентаризаций. Усовершенствование позволяет отслеживать период между инвентаризациями торговых точек и блокировать их по заданному графику;

– модуль контроля. Модуль позволяет руководителю организации осуществлять управление и контроль действий пользователя посредством заявок.

**Н.С. Седых (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **А.В. Воруев**, канд. техн. наук, доцент

## **SEO-АНАЛИЗ САЙТА УО «ГГУ ИМ. Ф. СКОРИНЫ»**

SEO – комплекс мер по поднятию позиций сайта в результатах выдачи поисковых систем по определенным запросам пользователей с целью продвижения сайта. При анализе эффективности поисковой оптимизации оценивается стоимость целевого посетителя с учётом времени вывода сайта на указанные позиции и конверсии сайта.

Для SEO-оптимизации используются SEO-инструменты, которые можно разделить на следующие виды:

– *инструменты анализа ключевых слов:*

1. Планировщик ключевых слов AdWords можно использовать для поиска вариантов ключевых слов и групп объявлений, оценки эффективности существующих ключевых слов, создания нового списка ключевых слов путем объединения нескольких списков, также помогает подбирать конкурентные ставки и планировать бюджеты кампаний.

2. KeywordTool.io применяется для исследования ключевых слов, используя функцию Автозаполнение Google.

3. UbbberSuggest извлекает ключевые фразы и предлагает ключевые слова, затем выводит их в виде списка в формате, удобном для копирования и вставки.

4. BuzzStream Meta Tag Extractor извлекает теги заголовков, ключевые слова и мета описания из списка URL.

5. Keyword Eye – визуальный инструмент по подбору ключевых слов для компаний в области контекстной рекламы (PPC) и SEO. С его помощью можно проанализировать ключевые слова конкурентов и данные об их ранжировании, определить частотные поисковые запросы в интересующей области деятельности, провести анализ трендов и ключевых слов, встречающихся на сайтах из топ-выдачи;

– *инструменты анализа контента:*

1. Copyscape ищет неподтверждённые копии страниц сайта в Интернете.

2. Siteliner генерирует отчеты о дублях, неработающих ссылках, валидность файла Sitemap.xml, PageRank внутренних страниц и др.

3. SEO-Browser покажет, как выглядит страница сайта для поисковых систем без графики и других элементов, которые они не могут распознать. SEO-Browser предоставляет информацию о метаданных, файле robot.txt, элементах JavaScript, внешних и внутренних ссылках сайта, соотношении качества текста к весу страницы.

– *технические инструменты:*

1. Google Webmaster Tools позволяет исследовать поисковые запросы, проверять правильность кода HTML, добавлять на сайт структурированную разметку данных и др. Кроме того, можно использовать этот инструмент, чтобы посмотреть ключевые слова, статистику изменений трафика, информацию о скорости загрузки сайта и т. д.

2. ScreamingFrog – отличный инструмент для анализа поисковой оптимизации сайта. Он быстро анализирует, проверяет и проводит обзор сайта с точки зрения SEO. Этот инструмент полезен для анализа крупных сайтов, которые сложно проверить вручную.

3. Advanced Meta Tag Generator & Google Search Results Preview покажет, как сайт представлен в результатах поисковой выдаче Google.

4. Google Developers PageSpeed Insights измеряет скорость загрузки веб-страниц как на стационарных ПК, так и на мобильных устройствах.

5. GTmetrix анализирует работу веб-страницы, используя данные Google PageSpeed и Yahoo! YSlow. Инструмент также предлагает рекомендации для решения обнаруженных проблем.

6. Pingdom покажет время загрузки всех страниц на сайте. Этот инструмент также помогает найти, что именно замедляет загрузку сайта. Pingdom также предоставит для анализа шкалу производительности, сообщит общее время её загрузки, размер страницы и рекомендации по ускорению вашего сайта.

7. Feedthebot – коллекция инструментов для анализа скорости отклика страницы и ее производительности. Включает инструменты: SEO Page Spider, Image SEO Tool, Google Guideline Check, Mobile SEO Tool, Social Media Count, Page Speed Tool, CSS Delivery Tool.

8. RobotsGenerator.com позволит быстро создать файл robots.txt.

9. Robots.txt Tester проверяет, действительно ли файл robot.txt не позволяет роботу Googlebot сканировать определенные страницы сайта.

10. Schema Creator автоматически генерирует микроразметку сайта в соответствии со стандартами Schema.org.

11. Structured Data Testing Tool является частью Инструмента для веб-мастеров Google, позволяет протестировать разметку на странице и проверить, может ли Google извлечь структурированные данные из нее.

12. XML-Sitemaps.com позволяет создать карту сайта.



13. XML Sitemap Inspector проверяет саму карту сайта и работоспособность ссылок карты.

С помощью некоторых из перечисленных инструментов был проанализирован сайт gsu.by. Используя BuzzStream Meta Tag Extractor было получено, что ключевыми словами главной страницы являются: Gomel, University, Scaryna, Skaryna, Scorina, Skorina, Scarina, Skarina, education, research, student, study. После этого в Планировщике ключевых слов AdWords была получена статистика запросов для полученного списка ключевых слов за последние 12 месяцев. Результат представлен на рисунке 1.

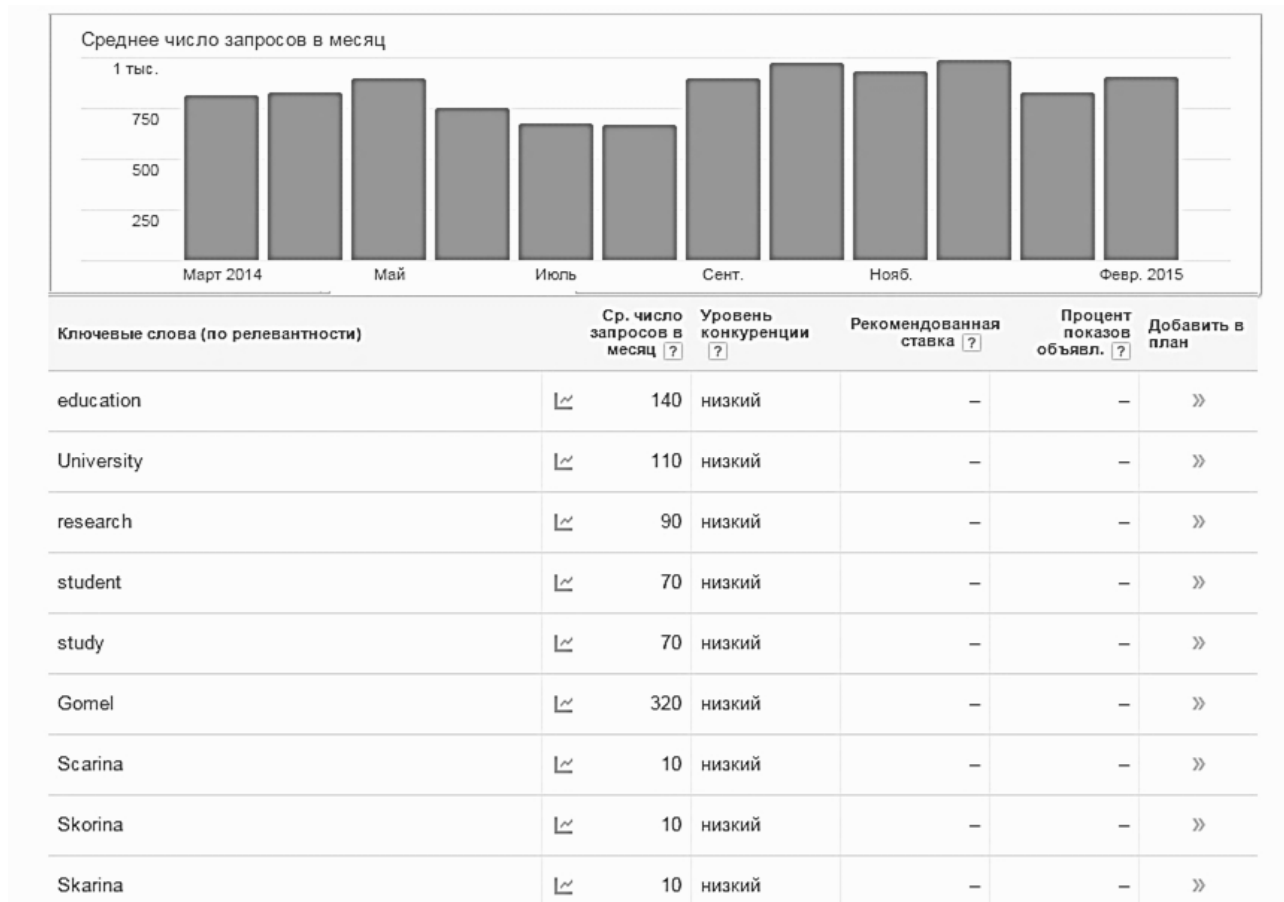
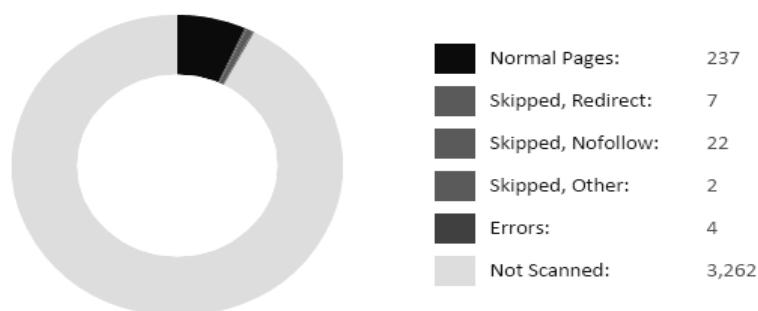


Рисунок 1 – Статистика запросов

Инструментом Siteliner было проанализировано 250 страниц из 3534 на дубли, неработающие ссылки и валидность ссылок. Результат анализа представлен на рисунке 2.

Также сайт был проанализирован на выполнение рекомендаций поисковых систем инструментом Feedthebot. Было получено, что карта сайта для поисковых систем и посетителей не соответствует рекомендациям. Также были выданы советы: повысить скорость загрузки страниц, добавить платные ссылки, сменить сервер.

Your Pages - Click below to see detailed results for your pages:



Your Duplicate Content - Click below to see your duplicate content:

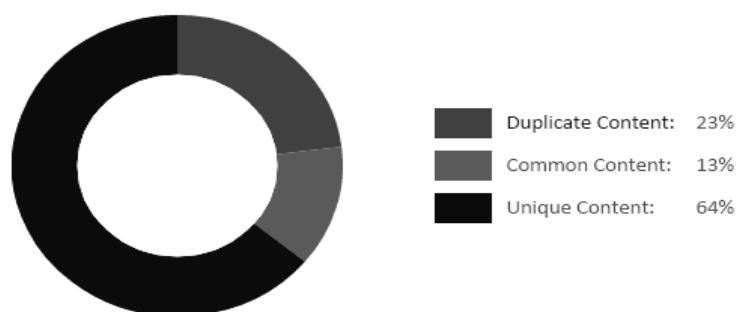


Рисунок 2 – Анализ сайта на дубли и неработающие ссылки

**А.А. Секацкий (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **П.Л. Чечет**, канд. техн. наук, доцент

## КРИТЕРИИ ВЫБОРА CMS

На данный момент на рынке представлено множество систем класса CMS. Способных решить практически любые задачи. Но, не смотря на это, часто при реализации Интернет проектов прибегают к созданию «своей» уникальной CMS . Такой подход нельзя назвать рациональным, возможно, он подойдёт для проектов с действительно уникальным функционалом. Но таких проектов, фактически мало. Обычно любой сложный функционал можно подразделить на несколько более простых, стандартных и распространенных. Поэтому, лучше остановиться на уже готовой платной или бесплатной CMS системе, поддержку которой осуществляет несколько компаний.

Подбор CMS следует осуществлять исходя из следующих критериев:

1. Количество компаний, осуществляющих техническую поддержку

Чем их больше, тем выше между ними конкуренция, что, несомненно, хорошо для потребителя.

## 2. Качество этих компаний

Желательно что бы тех. поддержку осуществляли компании (лучше крупные и известные), а не фрилансеры, что не удобно по многим причинам. К тому же хорошо когда существует какая-либо градация по квалификации – сертифицированный партнёр, золотой, платиновый и т. д.

## 3. Функциональность

Современная CMS система, не прибегая к программированию должна позволять:

- редактировать контент страниц, включая добавление/удаление графики;
- добавлять новые страницы;
- изменять структуру сайта и различные мета данные;
- настраивать регистрационные формы;
- управлять опросами, голосованиями и форумами;
- вести статистику посещений;
- распределять права по управлению сайтом среди пользователей.

## 4. Безопасность

После функциональности это, пожалуй, второй по важности критерий. Надо учитывать как надёжность системы со стороны внешних атак, так и от неосторожных действий пользователей системы.

## 5. Обновления

Не маловажным преимуществом готовых CMS систем над «своими» является именно факт их развития разработчиком без участия клиента и возможность обновления. Мало того, что таким образом можно получить функционал самостоятельно его не разрабатывая, но к тому же это позволит подсказать способ решения различных новых задач.

## 6. Гибкость

Гибкость системы определяется трудоёмкостью и необходимой квалификацией сотрудников для адаптации системы под свои требования – дизайн, структура, правовая политика и др.

## 7. Стоимость

При оценке стоимости, надо учитывать так же, сколько сайтов можно сделать на одной копии системы, какова стоимость тех. поддержки и последующих обновлений.

## 8. Технические требования

Необходимо чтобы система не требовала специфических настроек хостинга, которые может предоставить очень ограниченное число хостинг провайдеров. Очень важный критерий – потребность в ресурсах системы, её «жадность» к месту на диске, процессорной частоте и оперативной памяти. Если первое не особо важно, и при этом его не составляет особого труда узнать, то вторые два наоборот крайне важны и найти достоверные сведения о них крайне трудно.

## 9. Документация

Во многом успех от использования той или иной системы зависит от того, насколько легко в ней будет разобраться пользователям, что определяется наличием или отсутствием документации, справочной и учебной литературы и, конечно же, необходимым уровнем их технической квалификации.

**А.А. Секацкий (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **П.Л. Чечет**, канд. техн. наук, доцент

### **ЗНАЧЕНИЕ ИНТЕРНЕТА И WEB-САЙТА ДЛЯ ПРЕДПРИЯТИЯ КАК МАРКЕТИНГОВЫЙ ХОД**

Появление и развитие глобальной сети Интернет создало предпосылки для внедрения в деловую практику хозяйствующих субъектов многих стран мира новых возможностей для получения информации и ведения бизнеса электронным способом, особенно в сфере торговли. Основные ресурсы сети, такие как Всемирная паутина, электронная почта, позволяют партнерам устанавливать и поддерживать контакты независимо от их географического расположения и времени суток. Благодаря использованию различных поисковых систем, каталогов, баз данных, тематических и информационных сайтов намного упростился процесс поиска и получения информации о партнере, выпускаемой им продукции. Большинство западных компаний уже давно используют сеть Интернет для осуществления закупок и сбыта товаров и услуг на мировом рынке ввиду низких издержек этого канала.

Эффективной мерой продвижения товаров и услуг на внешние и внутренние рынки является подключение предприятий к глобальным сетям. Немаловажная роль при этом принадлежит web-сайту предприятия.

Именно его создание в сети Интернет позволяет компаниям работать на современном международном уровне.

Сайт предприятия в сети Интернет выполняет несколько основных функций:

- презентационная;
- коммуникативная;
- интерактивная;
- маркетинговая;
- инструмент для заключения сделок с партнерами и т. д.

Основные задачи, решаемые предприятиями при создании сайта:

– презентация предприятия, формирование благоприятного имиджа и продвижение брэнда предприятия, в том числе на новых региональных географически удаленных рынках;

– информационное обеспечение процесса принятия клиентом решения о заказе товара/услуги, распространение разносторонней информации о предприятии, его продукции и услугах (технические характеристики, отзывы предыдущих покупателей и т. д.);

– реализация продукции через Интернет: возможность оформления заявок на приобретение продукции, заключение договора электронным способом, создание на сайте электронных торговых площадок в форме бирж, тендеров и т. д.

– работа с клиентами и партнерами посредством электронных каталогов продукции, послепродажное обслуживание;

– проведение маркетинговых исследований: поиск информации, ее получение, обработка, анализ посещения сайта, проведение опросов, анкетирование потребителей, выявление тенденций сбыта продукции после начала новой рекламной кампании и т. д.;

– поддержка традиционной рекламной кампании в прессе, на радио, телевидении, прямой рассылки: во всех рекламных сообщениях указывается адрес сайта, на котором можно найти дополнительную информацию о предприятии, выпускаемой продукции, а также использование баннерной и контекстной рекламы.

Информационное наполнение сайта должно привлекать внимание посетителя и отвечать теме сайта. Кроме того, его форма должна соответствовать аудитории – материалы должны быть ясны, кратки и действенны в сети Интернет.

Структура и навигационные функции характеризуют организацию информации на сайте и возможности перемещения между его разделами. Хорошая структура и навигация – это признаки эффективности и организованности сайта. Они позволяют пользователю сформировать мысленную модель представленной информации, определить, где находятся необходимые сведения. Хорошие навигационные возможности позволяют быстро добраться до нужного места и легко охватить содержание сайта как вглубь, так и вширь.

Структура сайта должна включать:

1. Информацию о предприятии: история предприятия, достижения (награды, дипломы, сертификаты и т. д.), технологии, информация о партнерах.

2. Информацию о производимой продукции (услугах): каталог продукции (услуг); по возможности персональную страницу продукта (услуги); дополнительную информацию для заключения сделки; информацию о товаропроводящей сети; форму для заказа.

3. Наличие обратной связи. Наличие контактной информации для связи с производителем является одной из основных функций веб-сайта. Очень важно, чтобы все возможные контактные реквизиты были расположены в доступном разделе, причем ссылка на контактную информацию должна дублироваться на каждой странице сайта для облегчения контактов с потенциальными заказчиками.

Что касается дизайна сайта, то он представляет собой характеристику его внешнего вида. Сайт должен быть симпатичной домашней страничкой, хотя и не обязан быть ультрасовременным или суперстильным. Критерии оценки визуального оформления – высокое качество, уместность и соответствие той аудитории и задачам, на которые ориентирован сайт. Оформление должно производить впечатление на посетителя или даже поражать его.

В конечном счете, сайт – это нечто большее, чем просто сумма составляющих его частей. Помимо информационного наполнения, структуры и навигационных возможностей, визуального оформления, функциональности и интерактивности можно говорить об общем впечатлении о сайте. Общее впечатление учитывает и те неуловимые тонкости сайта, которые заставляют пользователя остаться или уйти.

Таким образом, подключение белорусских предприятий к международным сетям даст им следующие преимущества:

- доступ к оперативной информации для анализа и прогнозирования конъюнктуры зарубежных рынков из широко известных международных коммерческих автоматизированных банков данных, а также к Национальной компьютерной сети коммерческой информации;
- получение оперативной информации о потребностях в товарах и услугах;
- предоставление за рубеж оперативной информации о собственных коммерческих предложениях;
- снижение затрат при заключении договоров путем ведения электронных переговоров и т. д.

Кроме того, создание собственного сайта в сети Интернет открывает перед предприятием следующие возможности:

- создание яркого мультимедийного образа компании;
- создание базы данных заказчиков (при общении с заказчиками через сайт их обслуживание становится более комплексным);
- продажа продукции и услуг через сайт, а также осуществление платежей;
- создание базы данных поставщиков;
- создание центра реагирования на запросы клиентов;
- отслеживание статистики посещений сайта.

**Е.А. Сериков (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **В.А. Дробышевский**, ст. преподаватель

## **АВТОМАТИЗАЦИЯ УПРАВЛЕНИЯ РЕКЛАМОЙ НА ПЛАТЕЖНО-СПРАВОЧНЫХ ТЕРМИНАЛАХ РУП «БЕЛПОЧТА»**

В настоящее время актуальной темой является предоставление услуг в Интернете. На сегодняшний момент с помощью Интернета можно заказать различные товары с доставкой, оплатить услуги и т. д. В данном контексте будет рассмотрена оплата услуг через платежно-справочные терминалы (далее ПСТ) РУП Белпочта, так как большая часть населения, в силу привычки, делает различные платежи через банки и сберкассы. Однако посетители часто не догадываются о наличии ПСТ в данном отделении РУП Белпочта, либо просто забывают про них. Таким образом, эффективность ПСТ уменьшается, так как они были рассчитаны на то, что уменьшат нагрузку на операторов связи и уменьшат живую очередь. Для разрешения данной проблемы было решено создать программу, которая периодически оповещала бы посетителей отделения почтовой связи о том, что можно быстро оплатить услуги через ПСТ, не ожидая очереди.

Оплата услуг осуществляется посредством ПСТ производства ОАО «Туссон» и ОАО «Промсвязь», на которых установлено специализированное программное обеспечение. ПСТ подключены к серверу ПСТ, сопровождаемому техническим специалистом РУП «Белпочта».

Виды ПСТ представлены на рисунке 1 и 2.

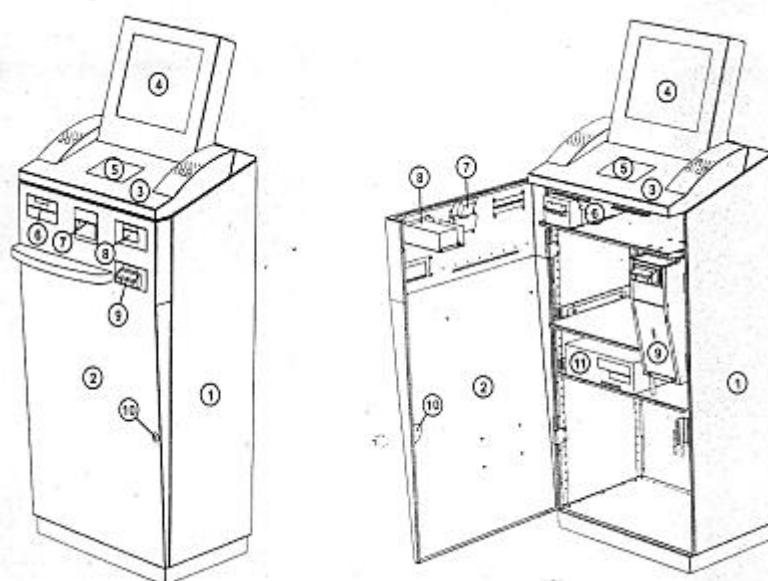


Рисунок 1 – ПСТ производства ОАО «Туссон»



Рисунок 2 – ПСТ производства ОАО «Промсвязь»

В состав ПСТ входят стандартные периферийные устройства, которые могут применяться в различных вариантах (по маркам и изготовителям периферийных устройств). Техническая составляющая практически идентична персональному компьютеру. Единственное отличие является то, что периферийные устройства подключаются через COM-порты. В состав таких устройств входят PIN-клавиатура, моторизованное устройство чтения-записи чиповых и магнитных карт, купюроприемник, чековый принтер. Конструкция предусматривает возможность подключения других устройств.

Все ПСТ работают под управлением ОС Windows XP.

Для решения поставленной задачи было решено создать программу голосовой и графической рекламы. Суть программы заключается в том, что с некоторой периодичностью воспроизводится звуковая дорожка, которая информирует и приглашает людей воспользоваться ПСТ для оплаты своих услуг. Помимо этого, во время простоя ПСТ на экране воспроизводится слайд-шоу различной информативности.

Программа имеет достаточное количество настроек для более гибкого использования. Так, например, можно регулировать громкость звука, период повторения как звуковой, так и графической рекламы, выбор аудиофайлов, которые будут воспроизводиться, выбор слайдов и их визуальных эффектов, период календарных дней, в которые будет проигрываться реклама. Этот пункт сделан по той причине, что в почтовых отделениях есть период времени пиковой активности платежей, то есть выдача пенсий, сроки оплаты коммунальных платежей и т. д. В другие дни особой активности и насыщенности в почтовых



отделениях не наблюдается, соответственно и необходимость в воспроизведении рекламы нет.

Помимо функции рекламы в программе реализован контроль над основным приложением ПСТ. Программа ведет мониторинг основного приложения и в случае его некорректного завершения блокирует работу ПСТ, выводя соответствующее сообщение на экран монитора. Во время блокировки ПСТ программа самостоятельно пытается перезагрузить основное приложение; период попыток задается в настройках программы так же, как и путь к контролируемому приложению. Если попытки остаются безуспешными, то к ПСТ выезжает специалист, который уже вручную произведет всю настройку ПСТ. Доступ к терминалу можно будет получить, введя пароль через PIN-клавиатуру.

Программа ведет журнал работы ПСТ, начиная с включения / выключения самого ПСТ, заканчивая всеми внештатными ситуациями с подробным указанием причины возникновения. Файлы хранятся на самом ПСТ.

Разработанная программа увеличила популярность использования платежно-справочных терминалов, сократила длину живой очереди, тем самым снизив нагрузку на операторов связи.

**Е.А. Симанкова (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.А. Дробышевский**, ст. преподаватель

## **ГРАФИЧЕСКОЕ ИССЛЕДОВАНИЕ И КЛАССИФИКАЦИЯ ПО ТИПУ ПРОЕКЦИЙ ОТОБРАЖЕНИЯ ПАНОРАМНЫХ ФОТОГРАФИЙ**

Проекция изображения происходит всякий раз, когда изогнутое изображение отображается на плоскости или наоборот. В программах по сборке панорам плоские панорамы так или иначе представлены в одной из проекций, некоторые из которых используются в картографии. Если посмотреть географические атласы, можно увидеть, как по-разному отображается сферический глобус на картах мира. В панорамной фотографии проекция сферы на плоскость аналогична картографическим проекциям. Проекция осуществляется, когда картограф отображает сферический глобус Земли на плоском листе бумаги, например. Поскольку полное поле зрения вокруг нас может рассматриваться как поверхность сферы (для всех углов зрения), для фотографий, которые будут показаны на плоском мониторе или отпечатаны, требуется аналогичная проекция сферы в плоскость.

В программах генерации панорам из отдельных связанных кадров можно собирать панорамы в различных проекциях. А тип проекции выбирается исходя из сюжета, задач и конкретного представления панорамы – полной или частичной.

Прямолинейную проекцию (часто ее называют проекцией «кастельной плоскости») можно представить в виде теней на плоском листе бумаги, который касается сферы в одной точке. Источник света при этом расположен в центре сферы [1].

Эквидистантная (сферическая) проекция отображает координаты широты и долготы сферического глобуса непосредственно на горизонтальные и вертикальные координаты сетки, где сетка имеет ширину примерно вдвое больше высоты. Горизонтальное растяжение, как следствие, усиливается по направлению к полюсам, так что северный и южный полюсы оказываются растянуты на всю верхнюю и нижнюю границы плоской сетки, соответственно. Эквидистантные проекции могут показать полный вертикальный и горизонтальный углы вплоть до 360 градусов.

Цилиндрическая проекция изображения аналогична эквидистантной, за исключением того, что по мере приближения к северному и южному полюсам объекты также растягиваются по вертикали так что на полюсах достигается бесконечное растяжение по вертикали (так что горизонтальная линия наверху и внизу плоской сетки отсутствует). Именно по этой причине цилиндрические проекции непригодны для изображений с большим вертикальным углом зрения. Цилиндрические проекции также являются стандартным типом, отображаемым традиционными панорамными плёночными камерами с поворотным объективом. Цилиндрические проекции сохраняют более точные относительные размеры объектов, чем прямоугольные, однако достигается это за счёт искривления линий, параллельных линии зрения (которые иначе оставались бы прямыми).

Прямоугольная проекция изображения имеет основное преимущество в том, что отображает прямые линии в трёхмерном пространстве в прямые линии на плоской двумерной сетке. Этот тип проекции соответствует тому, который создаёт большинство обычных широкоугольных объективов, так что она, вероятно, является наиболее понятной. Её основной недостаток состоит в том, что она может существенно преувеличить перспективу по мере увеличения угла обзора, что приводит к видимому завалу объектов к границам кадра. Именно по этой причине прямоугольные проекции обычно не рекомендуются для углов зрения, которые существенно превышают 120 градусов.

Рыбий глаз (fisheye) – это проекция изображения, целью которой является создание плоской сетки, где расстояние от центра сетки

примерно пропорционально действительному углу зрения; она образует изображение, которое выглядит похоже на отражение от металлической сферы. Как правило такая проекция не используется в качестве выходного формата панорамной фотографии, но вместо того она может представлять исходные изображения, если для съёмки таковых использовался объектив типа «рыбий глаз». Эта проекция к тому же ограничена вертикальным и горизонтальным углом обзора 180 градусов или менее, порождая изображение, которое помещается в круг. Её характеризует нарастающее искривление линий (которые иначе были бы прямыми) по мере удаления от центра изображения. Камера с объективом типа «рыбий глаз» исключительно полезна при создании панорам, которые покрывают всю сферу зрения, поскольку достаточно будет собрать небольшое число снимков.

Проекция Меркатора наиболее близко соотносится с цилиндрической и эквидистантной проекциями; она является компромиссом между этими двумя типами, обеспечивая меньшее растяжение по вертикали и более широкий употребимый угол зрения, чем цилиндрическая проекция, но с более сильным искривлением линий. Эта проекция, вероятно, является наиболее узнаваемой, поскольку используется в плоских картах мира. Отметим также, что альтернативная форма этой проекции (поперечный Меркатор) может использоваться для вертикальных панорам большой высоты.

Синусоидальная проекция изображения пытается сохранить равные площади во всех участках сетки. Если развернуть глобус в плоскость, можно вообразить, что такую проекцию можно свернуть обратно, чтобы сформировать сферу, которая будет идентична исходной по форме и площади поверхности. Характеристика равной площади полезна, поскольку если записывать плоскую проекцию сферического изображения, она сохранит неизменное горизонтальное и вертикальное разрешение по всему изображению. Эта проекция подобна рыбьему глазу и стереографической, за вычетом того, что сохраняет абсолютно горизонтальные линии из исходной сферы.

Стереографическая проекция очень похожа на рыбий глаз, но при этом сохраняет лучшее ощущение перспективы, увеличивая растяжение объектов по мере их удаления от точки перспективы. Подобная выделяющая перспективу характеристика в чём-то похожа на прямоугольную проекцию, хотя здесь она менее выражена.

## Литература

1. Ефремов, А.А. Панорамная фотография. Полноцветное издание / А.А. Ефремов. – СПб.: «Питер», 2012. – 128 с.

**Е.А. Симанкова (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **В.А. Дробышевский**, ст. преподаватель

## **ЭФФЕКТ ПАРАЛЛАКСА ПРИ ОПРЕДЕЛЕНИИ НОДАЛЬНОЙ ТОЧКИ ОПТИЧЕСКОЙ СИСТЕМЫ**

В процессе панорамной съемки могут возникнуть сложности с последующей стыковкой фотоснимков. Это происходит из-за эффекта параллакса. По определению, параллакс – это изменение визуального расположения объекта по отношению к удаленному фону в зависимости от того, как расположен наблюдатель.

Наглядно увидеть этот эффект возможно, если посмотреть сначала одним глазом, а потом другим – картинка изменится, произойдет смещение объектов. Особенно это заметно на объектах ближнего плана. Влияние параллакса на снимок при повороте камеры очевидно: происходит смещение объектов ближнего и дальнего плана относительно друг друга. Чтобы этого не происходило, необходимо вращать фотоаппарат вокруг специальной точки, называемой нодальной.

Нодальная точка – это точка на оптической оси объектива в месте пересечения лучей. При вращении камеры вокруг нодальной точки смещения объектов ближнего и дальнего плана не происходит. Расположение этой нодальной точки индивидуально для каждого объектива. Базовым приспособлением, позволяющим выполнять такое специфическое вращение, является панорамная головка, накручивающаяся на штатив. Такие головки могут быть универсальными, то есть подходят для создания как цилиндрических, так и сферических панорам. Нередко встречаются и такие модели, которые позволяют камере вращаться только в горизонтальной плоскости – они подходят только для создания цилиндрических панорам.

Ниже приведены определения параллакса, описанные в книге А.А. Ефремова «Панорамная фотография», которые различают и уточняют рассматриваемый эффект.

Параллакс видоискателя – несовпадение изображения, видимого в оптическом незеркальном видоискателе, с изображением, получаемым на фотографии. Явление параллакса, присущее определённым видоискателям, связано с тем, что оптические оси объектива и видоискателя не совпадают. При съёмке удалённых объектов это не очень заметно. Но при съёмке с близкого расстояния параллакс будет заметен, и чем ближе камера к объекту съёмки, тем сильнее будет виден сдвиг.

Временной параллакс – искажение формы объекта параллаксом, возникающим при съёмке фотоаппаратом со шторным затвором или

панорамной камерой с вращающимся объективом. Поскольку экспозиция происходит не одновременно по всей площади светочувствительного элемента, а последовательно по мере движения щели или вращения объектива, при съёмке быстро движущихся объектов их форма может искажаться. Например, если объект движется в ту же сторону, что и щель затвора, его изображение будет растянуто, а если в обратную, то сжато.

Параллакс панорамы – несовпадение одинаковых элементов изображения в перекрывающихся друг друга общих областях соседних кадров.

**Е.В. Скаковский (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. Е.А. Левчук, канд. техн. наук, доцент

## **РАЗРАБОТКА ПРОЕКТА КОНТРОЛЯ ТЕМПЕРАТУРЫ И УПРАВЛЕНИЯ ЗАДВИЖКАМИ НА БАЗЕ КОНТРОЛЛЕРОВ SIEMENS**

Программируемые логические контроллеры (ПЛК) фирмы Siemens – это промышленные контроллеры, используемые для автоматизации технологических процессов. На нашем рынке они используются для управления практически всеми узлами и агрегатами мини-ТЭЦ. Наиболее популярная линейка ПЛК Simatic S7 содержит семейства S7-200, S7-300, S7-400 и S7-1200. Все они включает в себя:

- модуль центрального процессора (CPU);
- блоки питания (PS) для питания контроллера от сети переменного или постоянного тока;
- сигнальные модули (SM), предназначенные для ввода/вывода дискретных и аналоговых сигналов;
- коммуникационные процессоры (CP), выполняющие автономную обработку коммуникационных задач в промышленных сетях Profibus, Industrial Ethernet и др.;
- функциональные модули (FM), которые выполняют задачи автоматического регулирования, позиционирования и пр.;
- интерфейсные модули (IM) для подключения стоек расширения к базовому блоку контроллера.

Кроме этого, к ПЛК через сеть Profibus подключалось большое количество ведомых устройств, таких как частотные преобразователи и прочие.

Главной утилитой является Step 7 – Simatic Manager, которая позволяет производить конфигурацию ПЛК и сетей (утилиты HWConfig и NetPro).

В процессе конфигурации определяется состав оборудования, способы подключения, используемые сети, адреса, выбираются настройки для используемых модулей. Готовая конфигурация загружается в ПЛК, что так же является настройкой оборудования. Утилиты конфигурации позволяют осуществлять диагностику оборудования, обнаруживать аппаратные ошибки или неправильный монтаж.

В докладе обсуждаются детали реализации системы контроля температуры и управления задвижками на базе контроллеров Siemens S300.

**Т.С. Скринникова (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Леванцов**, ст. преподаватель

## **СРАВНИТЕЛЬНЫЙ АНАЛИЗ СЕРВЕРНЫХ СУБД**

Из года в год на мировом рынке СУБД наблюдалась достаточно стабильная картина – с большим отрывом лидировали несколько крупных компаний, оставляя других далеко позади.

Еще несколько лет назад тройка лидеров была неизменна. Так, согласно отчету Forrester, в 2009 г. едущую группу СУБД составляли решения трех компаний – Oracle Database, IBM DB2 и Microsoft SQL Server.

Однако стремительное развитие технологий и усиление конкуренции нарушили эту стабильность – стали активно заявлять о себе новые компании. Показательным в этом плане является сектор операционных СУБД – систем, которые подходят для обширного спектра транзакционных приложений корпоративного уровня. Отчет Gartner за 2013 г. представил в секторе лидеров четыре компании: три вышеупомянутых и SAP.

Четыре компании, которые в 2014 г. оказались в секторе лидеров рынка операционных СУБД несомненно, имеют много общего. Но при этом у каждой из них свои достоинства, свои недостатки и разные перспективы.

*Oracle Database.* Oracle Database предлагает новую мультиарендную архитектуру, которая упрощает быструю консолидацию большого числа баз данных и управление ими как облачной службой. Oracle Database также предоставляет возможности обработки данных в памяти и беспрецедентную эффективность аналитики. Дополнительные инновации, реализованные в базах данных, обеспечивают эффективность, производительность, безопасность и доступность совершенно нового уровня. Oracle Database поставляется в трех выпусках, соответствующих

любым бизнес-нуждам и бюджету. Enterprise Edition, Standard Edition и Standard Edition One.

Основными плюсами являются:

- предварительно заданные настройки – быстрая установка на один сервер благодаря предварительным настройкам и готовой к использованию СУБД, и автоматизированным средствам управления;

- простая в использовании консоль управления, основанная на веб-браузере и обеспечивает возможность доступа и управления из любой точки, а также просмотра состояния в реальном времени;

- обработка различных форматов – поддерживает все типы стандартных реляционных данных и встроенную обработку данных текста, документов, аудио, видео и XML;

- автоматизированные средства резервного копирования – позволяют обходиться без дорогостоящих патентованных решений для хранения данных благодаря автоматизированным функциям зеркалирования, резервного копирования и восстановления;

- поддержка ролей – простое управление полномочиями и правами доступа к базам данных;

- совмещенные операции – управление задачами совмещения операций в условиях больших рабочих нагрузок;

- интеграция с Visual Studio – с помощью средств Oracle Developer Tools для Visual Studio можно легко разрабатывать приложения .NET на базе Oracle;

- совместимость с целым рядом операционных систем – исключительная производительность в среде Windows и других операционных системах, включая Linux и Unix, обеспечивает гибкость при выборе платформы.

Однако у компании есть проблемы в области маркетинга и видения рынка. Из минусов решений Oracle можно еще отметить привязку самих СУБД к некоторым проприетарным продуктам компании, высокую цену на решения, а также встречающиеся ошибки программирования. Слабый маркетинг и сложности с прогнозированием новых трендов играют на руку конкурентам, которые, используя эти пробелы Oracle, могут подтянуться к ней.

*SAP*. Еще один участник «большой четверки» – компания SAP. Компания занимается разработкой автоматизированных систем управления такими внутренними процессами предприятия, как: бухгалтерский учет, торговля, производство, финансы, управление персоналом, управление складами и т. д. Приложения обычно можно адаптировать под правовой контекст определенной страны. Кроме поставок программного обеспечения, фирма предлагает услуги по его внедрению,

используя для этого собственную методологию внедрения (изначальное название ASAP – Accelerated SAP, впоследствии – ValueSAP).

Ее несомненным плюсом следует назвать грамотный анализ и понимание рынка в сегменте ИТAP. SAP активно развивается во многих направлениях и продолжает лидировать на мобильном рынке, предлагая системы с максимально широким спектром функциональных возможностей. Еще один голос в пользу систем SAP вносят высокая производительность и простота использования решений.

Но, как и у Oracle, у SAP есть проблемы в области маркетинга. Еще один минус – нехватка специалистов по решениям SAP, а также есть проблемы с непрерывностью и сложностью ведения бизнеса. Не приспособлена к нестабильности отечественного законодательства: частым изменениям требований к оформлению отчетной документации и пр.

*IBM DB2.* Один из старейших игроков рынка СУБД – компания IBM. Ее сильными сторонами, позволяющими из года в год поддерживать статус лидера, являются высокая производительность решений и обширное присутствие на рынке. Система IBM имеет высокие показатели катастрофоустойчивости. Сама компания осуществляет поддержку решений на множестве вертикальных рынков и имеет одну из крупнейших сетей партнеров на всем секторе ИТ. Большим преимуществом является бесплатная версия без ограничений на размер базы и ограничителя запросов, а также бесплатная техподдержка.

Однако есть у компании и слабые места. Одно из них – малое количество специалистов и высокая стоимость хороших специалистов. Также из минусов можно отметить размер баз больше, чем в других СУБД, медленная загрузка dt-файла, понижение уровня логирования ошибок, для корректной обработки некоторых сообщений платформ.

Эти минусы, если не будут исправлены компанией, дают шанс другим перспективным игрокам подобраться ближе к «магической четверке».

*Microsoft SQL Server.* Microsoft SQL Server – сильная сторона компании Microsoft, а сама компания занимает второе место по общей доле выручки среди всех вошедших в исследование компаний. Клиенты высоко оценивают производительность, поддержку и простоту внедрения SQL Server.

Из плюсов можно выделить усовершенствованный процессор запросов, динамическое блокирование на уровне строк, динамическое самоадминистрирование, готовность к использованию в Интернете, интрасетях и для электронной коммерции, интеграция с другими продуктами Microsoft.

Однако есть трудности и в этой компании, ключевая проблема – восприятие ее решений в бизнес-среде. Клиенты не используют решения Microsoft для создания корпоративных критически важных приложений,



и это тормозит дальнейшее распространение SQL Server в качестве основной корпоративной СУБД.

Каждая компания имеет свои достоинства и недостатки. У них практически одинаковые свойства и возможности. Поэтому, выбор продукта зависит от вкусов и возможностей предприятия или пользователя.

**Т.С. Скринникова (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Кулинченко**, ст. преподаватель

## **НОВИНКА ОТ GOOGLE – МОДУЛЬНЫЙ СМАРТФОН GOOGLE ARA PROJECT**

Модульный смартфон Google Ara Project – новая широкомасштабная инициатива Google в сфере мобильных технологий. Компания разрабатывает платформу со свободным аппаратным обеспечением и системной рамой, на которую каждый пользователь сможет самостоятельно, без труда и специальных инструментов установить дополнительные модули, будь то батарея, дисплей, клавиатура или любой другой модуль, который нужен конкретному пользователю. Инновационный модульный телефон Google призван заменить стандартные устройства, которые невозможно модернизировать.

Обзор технических характеристик новинки позволяет составить объективное представление об аппарате. Размер стандартного телефона будет средним, ориентировочно между размерами Galaxy S5 и iPhone. Командой разработчиков Project Ara была создана платформа, которая имеет три форм-фактора, одну из которых можно подобрать согласно своим вкусовым предпочтениям и желаниям. Так, одна из них – (mini) – маленькая и компактная, другая – (medium) – средняя, и последняя – (jumbo), самая крупная, созданная для фаблетов.

Модули будут выполнены в виде плиток на электропостоянных магнитах для подключения к платформе. Несмотря на отсутствие тока, они смогут сохранять намагниченность. Такая технология, называемая UniPro, и станет основой для подключения модулей к основному корпусу модульного смартфона.

Но есть и базовые требования, которые не изменяются и меняться не будут. К примеру, основа, или корпус, будет прямоугольной и создана из 6061 алюминия.

Будут доступны и конфигурации с применением внешних модулей, например, модуль определения кислорода в крови.

Проект был изначально возглавляем командой разработчиков в Motorola Mobility – дочерней компании Google. И хотя Google продал

Motorola в Lenovo, он сохранил команду, работающую в этом направлении. Но при этом, платформа модульного смартфона Project Ara будет совершенно открытой, то есть поддерживать модули любого разработчика. Главным условием разработки модуля является использование MDK (Module Developers Kit). При этом модули можно будет менять без отключения питания мобильного устройства. Такая «горячая» замена позволит пользователю всегда оставаться на связи.

Модуль будет иметь широкие возможности, при этом выполнять разные функции. Например, будучи задним экраном, он сможет заменить функции крошечной батареи, чтобы максимально снизить расход энергии.

Проект модульного кастомного смартфона Project Ara можно сравнить с моделью Moto X, где пользователь также мог выбрать цветовую гамму задней крышки, кнопок, гравировку, объем памяти и фоновый рисунок. Подразумевается, что точно так же пользователь будет конфигурировать устройством Project Ara.

Планируется, что Ara-телефоны смогут активно работать на протяжении 5–6 лет, что и станет их преимуществом перед современными смартфонами. Это связано с тем, что обновлять свое устройство пользователь сможет постепенно, используя более мощный процессор или улучшенную камеру, а не дожидаться выхода нового мобильного устройства.

А купить модули можно будет в магазине компонентов для Project Ara – его в ближайшее время после выхода телефона создаст сама корпорация.

Выглядеть и работать данный магазин будет как Google Play Store. Каталог будет открытым, и все модули, загружаемые в каталог, также будут открытыми. Любой разработчик, создавший собственный модуль, после одобрения компании сможет предлагать такой компонент покупателям через централизованный каталог от Google.

Ранее Google объявила, что Rockchip станет одним из производителей компонентов для смартфонов Project Ara. Позднее Google сообщила, что к проекту присоединились ещё два технологических гиганта: Marvell и NVIDIA. Google и Quanta создали два референтных устройства: один на базе процессора Marvell PXA1928 и другой на базе NVIDIA Tegra K1. Последний уже зарекомендовал себя в других проектах, в частности, он установлен в планшете Google Nexus 9. К сожалению, какие-то технические характеристики и тактовая частота каждого из процессоров, использованных в Project Ara, не раскрываются разработчиками.

Если о процессоре Tegra K1 известно, что он построен на архитектуре Kepler, графическое ядро поддерживает не только OpenGL ES 3.0, но и OpenGL 4.4, DirectX 11 и CUDA 6. Четыре ядра Cortex-A15 с двумя

мегабайтами кэш-памяти второго уровня и пятым вспомогательным ядром-«компаньоном» (также Cortex-A15), предназначенным для работы в экономичном режиме с невысокой вычислительной нагрузкой. Модуль работы с камерами имеет мощный сдвоенный процессор обработки изображений (ISP), имеющий пропускную способность до 1,2 гигапикселя и поддерживающий сенсоры с разрешением до 100 мегапикселей. Модуль работы с устройствами вывода изображения поддерживает разрешение до UltraHD, также известное как 4K (что было и в Tegra 4, впрочем) и для встроенного дисплея и внешних, подключаемых по HDMI 1.4a, а вместе с этим чип поддерживает и декодирование данных в формате H.265 в этом разрешении.

Новый чип NVIDIA производится по 28-нанометровому «HPM» техпроцессу.

Чип от Marvell является загадкой. Он был представлен на выставке MWC 2014 и построен на архитектуре ARMv8. Процессор оснащён четырьмя ядрами Cortex-A53 с частотой 1,5 ГГц и графическим ускорителем Vivante GC5000, что обеспечивает поддержку 1080p-дисплеев и запись видео с аналогичным разрешением. Marvell PXA1928 поддерживает LTE Cat 4, TD-HSPA+, LTE TDD/FDD, HSPA+, EDGE, CSFB и VoLTE. Здесь же присутствует поддержка Wi-Fi 802.11ac, Bluetooth 4.0, FM-радио и NFC. Кроме того, процессор от Marvell поддерживает продвинутую систему мобильной безопасности, включая Root Key Protection, Secure Boot и другие.

Вопрос операционной системы, который был главной проблемой, был решен с помощью Google: Ara получит на борт модифицированную версию Android L.

Объединив Android с пакетом наработок от Linaro, некоммерческой организации, занимающейся разработкой и консолидацией ПО со свободным кодом, компания Google собирается представить версию своей операционной системы с полной поддержкой заменяемых элементов в устройстве без необходимости скачивания драйверов и обновлений для каждого нового элемента.

**А.С. Сметанко (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **В.А. Дробышевский**, ст. преподаватель

## **ОРГАНИЗАЦИЯ HTTP ЗАПРОСОВ С ПОМОЩЬЮ БИБЛИОТЕКИ VOLLEY**

Для каждого клиент-серверного приложения возникает вопрос о способе приема-передачи данных от сервера к мобильному клиенту.

Чаще всего в современной реализации на стороне сервера применяют REST, т. е. для получения или изменения данных необходимо «выполнить» модифицированный HTTP запрос с клиентского приложения. REST – единый стандарт взаимодействия клиентских приложений с сервером, который не требует со стороны сервера реализации и поддержки протокола для каждого нового клиента.

В мобильной OS Android выполнение HTTP запросов должно осуществляться не из основного потока приложения. В случае выполнения из основного потока возникает ошибка «Network OnMain Thread Exception». Для реализации запросов можно воспользоваться стандартным Java классом «Thread», но при попытке изменить свойства отображаемых на экране компонент приведет к ошибке «Called From Wrong Thread Exception» это связано с особенностью самой OS Android, в состав которой входят различные механизмы взаимодействия с основным потоком. Например, специальный класс «Handler» позволяет реализовать изменения компонент после выполнения HTTP запроса.

С версии 3 API SDK Android появился инструмент «AsyncTask», который взял на себя:

- выполнение метода «onPreExecute» в основном потоке приложения перед запуском отдельного потока;
- выполнение метода «doInBackground» в отдельном потоке;
- выполнение метода «onPostExecute» после работы отдельного потока, но уже в основном потоке приложения.

Однако проблему организации и выполнения HTTP запросов «AsyncTask» возложил на программистов.

В 2013 г. компания Google представила библиотеку «Volley», на конференции «Google I/O 2013». Библиотека делает сетевые приложения для Android проще и, самое главное, быстрее, но она не входит в основной состав SDK.

Преимущества использования «Volley»:

- обработкой и кэширование сетевых запросов;
- автоматическое составление всех сетевых запросов;
- прозрачность дискового кэширования и кэширования в памяти;

API для отмены запросов, можно отменить один запрос или установить несколько запросов для отмены;

– предоставляет мощные возможности изменения параметров запроса;

– инструменты отладки и трассировки;

– предоставляет инструменты для создания шаблонных ответов не только в виде класса «InputStream» но и в любом другом (например, String или JSONObject).

**В.В. Старовойтов (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **Е.А. Ружицкая**, канд. физ.-мат. наук, доцент

## **РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ТОРГОВЫХ ОПЕРАЦИЙ**

В настоящее время рынок мобильных устройств является самым быстрорастущий среди вычислительных устройств и оценивается в 1,6 млрд. устройств, в то время как рынок ПК насчитывает около 2 млрд. пользователей (по данным на конец 2013 г.). Современные мобильные устройства обладают внушительной вычислительной мощностью, множеством встроенных датчиков (системой глобального позиционирования, гироскоп, фотокамера и т. п.), поддерживают несколько сетевых протоколов и т. д. Сочетание вычислительной мощности, доступ к множеству встроенных сенсоров и легкость распространения приложений на рынке сделало мобильные устройства новой платформой для бизнеса.

Стремление повысить качество и скорость обслуживания покупателей заставило пересмотреть подходы в работе точек продаж. POS (англ. point of sale – точка продажи) – это место, где производится оплата товаров или услуг, подсчёт сдачи, печать чеков и др. Мобильная POS – это смартфон или планшет, выполняющий функции кассового аппарата и/или терминала оплаты.

Разработанное приложение представляет собой мобильную POS-систему и позволяет выполнять некоторые функции кассового аппарата и/или терминала оплаты, удобный интерфейс взаимодействия с пользователем облегчает поиск нужного товара и получение его характеристик. Приложение может взаимодействовать с внешними устройствами, такими как bluetooth-принтер, сканер пластиковых карт и штрих кодов.

Поиск товаров и оплата производится на стороне сервера. Приложение написано на языке Objective-C, в роли СУБД было решено использовать sqlite. Общение с сервером происходит с помощью RESTful API.

Особое внимание уделено защите персональных данных покупателей. Информация о кредитных и дебетовых картах передаётся на сервер в зашифрованном виде и не хранится на устройстве.

Таким образом, мобильное приложение позволяет отчасти заменить классический кассовый аппарат, улучшает процесс покупки и дает возможность увеличивать функциональность точки продажи без необходимости обновления аппаратных средств.

**Д.Е. Стасев, А.Е. Лисун**  
**(УО «ГГТУ имени П.О. Сухого», Гомель)**  
Науч. рук. **В.С. Мурашко**, ст. преподаватель

## **РАЗРАБОТКА И РЕАЛИЗАЦИЯ ОСНОВНЫХ ФОРМ И ЗАПРОСОВ БАЗЫ ДАННЫХ «РАСЧЕТ ПРИПУСКОВ НА МЕХАНИЧЕСКУЮ ОБРАБОТКУ»**

В настоящее время в любой области промышленности практически невозможно представить себе проектирование достаточно серьезных изделий без применения средств автоматизации проектирования. Наиболее широкое применение средств автоматизированного проектирования получили в области машиностроения. Поэтому решение проблем эффективного конструирования деталей и узлов механизмов, и разработка для этого средств повышения эффективности работы конструкторов были и остаются одним из наиболее перспективных направлений текущих работ в области САПР.

Необходимость в разработке инфологической (информационно-логической) модели, не зависящей от средств программной реализации хранения и обработки данных и отражающей интегрированные структуры данных предметной области («Расчет припусков на механическую обработку») появилась потому, что традиционное представление справочно-нормативных данных в [1] не пригодно для автоматизации расчета припусков на механическую обработку. Каждая таблица в [1] представляет собой совокупность нескольких таблиц (массивов данных), связанных между собой по какому-то признаку (признакам). Поэтому возникла первоначальная задача – проанализировать предметную область и установить логические связи между информационными объектами предметной области.

Одним из основных этапов проектирования технологического процесса изготовления детали является выбор вида заготовок, определение их размеров и припусков.

Припуск – слой материала, удаляемый с поверхности заготовки в целях достижения заданных свойств обрабатываемой поверхности детали.

Существуют следующие методы расчета припусков: опытно-статистический, расчетно-статистический и расчетно-аналитический [1].

Опытно-статистический и расчетно-статистический методы оперируют набором таблиц, по которым определяется величина припуска в зависимости от размеров поверхности, вида и метода обработки. Расчетно-аналитический метод базируется на анализе производственных

погрешностей и дифференцированно учитывает влияние на величину припуска конфигурации и размеров детали, качества заготовки, погрешностей механической и термической обработки. Вместо полной величины припуска определяются составляющие припуск элементы и устанавливается простая функциональная связь между ними.

Проанализировав предметную область «Расчет припуск на механическую обработку», был выбран расчетно-аналитический метод расчета припусков, описанный в [1].

Информацию, используемую при автоматизации расчета припусков и допусков условно можно разделить на исходную и производную. Исходной называется информация, существующая до начала машинного проектирования. Она подразделяется на переменную и условно-постоянную.

В качестве переменной информации можно отнести, например требуемая точность выполнения заготовки, шероховатость и качество ее поверхностных слоев и т. д. Эта информация вводится в оперативное запоминающее устройство каждый раз при новых расчетах припусков и допусков.

Условно-постоянная информация для расчета припусков состоит из нормативных таблиц [1]. Эта информация является достаточно стабильной и постоянно хранится во внешней памяти ЭВМ.

Производная информация формируется в процессе расчета припусков и допусков и содержит их значения.

В качестве информационных структур, необходимых при автоматизации расчета припусков и допусков были взяты следующие таблицы [1].

1. Порядок расчета припусков на обработку и предельных размеров по технологическим переходам.

2. Расчетные формулы для определения припуска на обработку.

3. Качество поверхности различных видов заготовок.

4. Качество торцевой поверхности после резки заготовок из горячекатанного проката.

5. Параметры, достигаемые после механической обработки наружных поверхностей.

6. Параметры, достигаемые после механической обработки отверстий.

7. Суммарное значение пространственных отклонений для различных видов заготовок и механической обработки.

8. Удельная кривизна заготовок на 1 мм длины.

9. Удельный увод и смещение оси отверстий при сверлении.

10. Погрешность закрепления заготовок при установке в радиальном направлении для обработки на станках.

11. Погрешность закрепления заготовок при установке в осевом направлении для обработки на станках.

12. Погрешность закрепления заготовок при установке на опорные штифты приспособлений.

13. Погрешность закрепления заготовок при установке на опорные пластинки приспособлений.

Следующая задача – это построение датологических моделей, ориентированных на среду хранения и обработки данных. Выбор средств реализации базы данных определяет вид этих моделей. В качестве рабочей среды была выбрана высокопроизводительная система управления реляционными базами данных Microsoft Access 2003.

Результаты решения второй задачи:

– создана база данных «Расчет припуск на механическую обработку»: разработана структура перечисленных выше таблиц и связи между ними;

– разработаны формы для ввода и выбора данных;

– разработаны удобные кнопочные формы. Кнопки такой формы обеспечивают вызов других кнопочных форм, а также отдельных объектов: отчетов, форм, макросов, с которых начинается решение задачи;

– разработаны все виды запросов к данным БД, необходимые в процессе расчета припусков;

– созданы формы – справки;

– разработаны макросы, позволяющие открывать, закрывать, отправлять на печать формы и отчеты;

– созданы отчеты для получения жесткой копии выбранной из БД информации.

Вызов главной кнопочной формы – панели управления приложением при открытии базы данных «Расчет припуск на механическую обработку» позволяет пользователю сразу начать работу в среде приложения и приступить к выполнению задачи расчета припуска.

Использовать предлагаемую систему для автоматизированного расчета припусков могут студенты в курсовых и дипломных работах, а также инженеры-технологи.

## Литература

1. Горбацевич А. Ф., Шкред В. А. Курсовое проектирование по технологии машиностроения: [Учеб. пособие для машиностроит. спец. вузов]. – 4-е изд., перераб. и доп. – Мн.: Выш. школа, 1983. – 255 с.



**В.Н. Стельченко (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **П.Л. Чечет**, канд. техн. наук, доцент

## **ОПИСАНИЕ ВОЗМОЖНОСТЕЙ СИСТЕМЫ УПРАВЛЕНИЯ САЙТОМ JOOMLA**

Основные характеристики Joomla!:

Полностью основанный на БД движок с использованием PHP/MySQL.

Модуль безопасности для многоуровневой аутентификации пользователей/администраторов.

Секции новостей, продукции или услуг легко редактируемы и управляемы.

Разделы тем могут быть добавлены авторами.

Полностью настраиваемые схемы расположения элементов, включая левый, правый и центральный блоки меню.

Закачка изображений при помощи браузера в вашу собственную библиотеку – для последующего использования с любого места сайта.

Форум/Опросы/Голосования для эффективной обратной связи.

Работа под Linux, FreeBSD, MacOSX, Solaris, AIX, SCO, Windows и Windows Server.

Возможности администрирования:

Возможность создавать неограниченное количество страниц.

Четкая организация структуры сайта.

Для каждой динамической страницы можно создать свое описание и ключевые слова в целях повышения рейтинга в поисковых системах.

Начало и окончание публикации любых материалов можно запрограммировать по календарю.

Возможность ограничить доступ к определенным разделам сайта только для зарегистрированных пользователей.

Секции новостей, продукции или услуг легко редактируемы и управляемы.

Полностью настраиваемые схемы расположения элементов по 5 областям, что позволяет сделать оригинальный дизайн.

Закачка изображений при помощи браузера в вашу собственную библиотеку – для последующего использования с любого места сайта.

Опросы и голосования для эффективной обратной связи.

Различные модули – такие как последние новости, счетчик посетителей, подробная статистика посещений, гостевая книга, форум и т. д., гибкая настройка, будут ли показаны эти модули и на каких страницах.

Возможность создания не одной, а нескольких форм обратной связи для каждого контакта.

Изменение порядка объектов, включая новости, статьи и т. д.

Генератор показа случайной новости.

Модуль приёма от удалённых авторов новостей, статей и ссылок.

Иерархия объектов – количество секций, разделов, подразделов и страниц, зависит от вашего желания.

Библиотека изображений позволит Вам хранить все Ваши GIF- и JPEG-файлы под рукой для лёгкого доступа.

Менеджер рассылки новостей. Выбирайте из более чем 360 служб рассылки новостей по всему миру.

Менеджер архива. Помещайте Ваши старые статьи в архив, чем просто удалять их.

Возможность распечатать или отправить другу на e-mail любую статью с сайта.

Выбор из 3 визуальных редакторов, что упрощает редактирование материалов до уровня редактирования текста в программе Word.

Предварительный просмотр перед окончательным размещением.

Возможность легкой смены дизайна.

Экономное использование места на сервере за счет использования базы данных MYSQL.

Возможность использования адресов страниц адаптированных для лучшей индексации всеми поисковыми системами.

На рисунке 1 в графическом виде представлен рейтинг платных и бесплатных CMS (англ. Content management system).

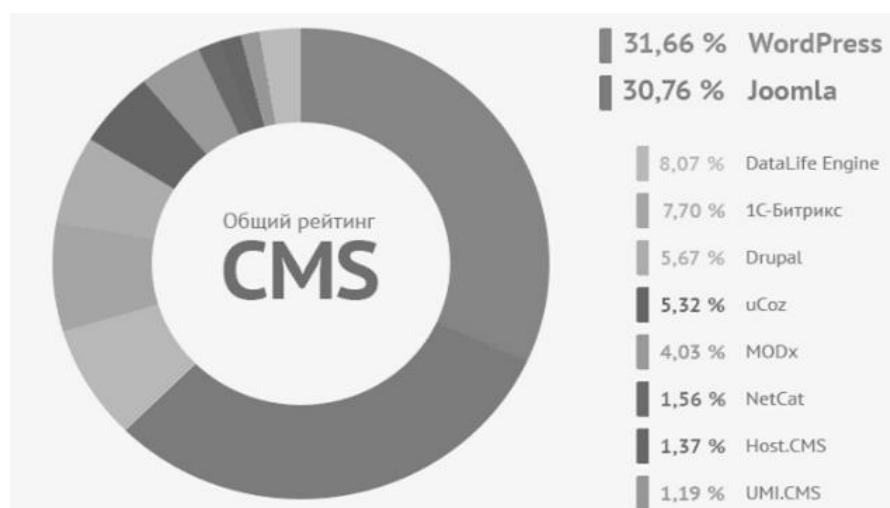


Рисунок 1 – Рейтинг платных и бесплатных CMS

Из рисунка 1 видно, что JOOMLA занимает лидирующую позицию по использованию.

**В.Н. Стельченко (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **П.Л. Чечет**, канд. техн. наук, доцент

## **ПРИМЕНЕНИЕ ИНТЕРНЕТ ТЕХНОЛОГИЙ В БИЗНЕСЕ**

В наше время, когда услугами интернета пользуется более 90 % населения, использование рекламы только в средствах массовой информации представляется не полным и недостаточно эффективным. Избалованный доступом к информации потребитель уже как данность воспринимает тот факт, что информацию о серьёзном товаре или услуге можно найти на сайте предприятия.

Также изжили себя сайты-визитки предприятия. Их наличие говорит больше не в пользу представленной организации, а наоборот, играет против неё. Если фирма не может позволить себе содержать хороший полноценный сайт с актуальным именем – она не может претендовать на доверие потенциальных клиентов.

Чтобы WEB-сайт соответствовал современным реалиям Интернет рынка, он должен соответствовать следующим параметрам:

- размещаться на полноценных хостингах;
- доменное имя должно быть второго уровня с понятным URL и в зоне нахождения головного офиса;
- не загружен графикой, анимацией flash, необходимо думать о клиентах, заходящих на сайт с ограничением по трафику;
- правильно подобранная цветовая гамма для удобства чтения;
- проведена SEO-оптимитизация;
- размещение прямых ссылок в профильных каталогах;
- удобный (интуитивный) интерфейс навигации;
- наличие фотогалереи с участием работников предприятия придает сайту живое лицо;
- контролирование формы обратной связи для оперативных ответов;
- наличие зеркала сайта;
- приветствуется размещение карт со схемой проезда к офису (предприятию) в сервисах Yandex или Google.

На первый взгляд выше перечисленные параметры требуют много ресурсов, однако, если придерживаться этих принципов, web-сайт станет надежным помощником в ведении бизнеса, и поможет повысить прибыль. Грамотно осуществлённое присутствие в сети Интернет помогает не только повысить удобство работы существующих клиентов с организацией, но и помогает привлечь новых клиентов, заявить о себе на новых рынках.

**А.В. Сюськин (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Кулинченко**, ст. преподаватель

## **АВТОМАТИЗАЦИЯ ТОРГОВОГО ОБЪЕКТА ПОД УПРАВЛЕНИЕМ СИСТЕМЫ 1С:ПРЕДПРИЯТИЕ**

В настоящее время невозможно представить торговый объект без соответствующего оборудования. В его состав входят кассовые модули, фискальные регистраторы, считывающее и прикладное оборудование, которое можно встретить, делая покупки в любом магазине. Слабое применение средств оргтехники приводит к снижению производительности труда и эффективности работы управленческого и технического персонала. Степень автоматизации рабочего места торгового объекта влияет на качество обслуживания покупателя, снижение ошибок в работе персонала, скорость обработки данных.

Автоматизация торгового объекта – это комплекс мероприятий по разработке и внедрению в торговый бизнес-процесс высокотехнологичного оборудования и программного обеспечения с целью повышения эффективности использования трудовых ресурсов и качество обслуживания. Кроме того, автоматизация торговли это экономия и увеличение рабочего времени за счет сокращения числа ревизий, возможность прослеживания всего пути товара и денежных средств, контроль остатка товара.

Выделяют аппаратную и программную составляющую автоматизации торгового объекта. К аппаратной части относят: кассовые POS-системы, оборудование штрих-кодирования, принтеры и сканеры штрих-кода, торговые весы с принтером этикеток, терминалы сбора данных, информационные киоски, электронные ценники. К программному обеспечению можно отнести: кассовые программы для управления POS-системой, кассовые средства для управления кассовым узлом, системы управления торговым залом, системы управления предприятием, дополнительные модули управления и программы для автоматизации магазинов.

Остановимся на системе управления предприятием, которая включает в себя соответствующее программное обеспечение и сканер штрих-кода. В качестве программного обеспечения выступает комплект 1С:Предприятие 7.7. Эта программа позволяет управлять предприятием в полном объеме и одновременно взаимодействовать как с торговым оборудованием, так и с системой управления торговым залом.

Главной задачей данного проекта является разработка модуля торговых операций на базе механизма взаимодействия торгового оборудования с системой управления предприятием 1С:Предприятие 7.7.

По мере появления магазинов и торговых площадок европейского типа в последние 3–4 года активно развивается новое направление автоматизации предприятий – системы для оптовой и розничной торговли. Эти системы не только полностью обеспечивают автоматизацию бухгалтерских задач, но и помогают осуществить оперативное управление деятельностью торгового предприятия. В то же время торговые системы существенно отличаются от бухгалтерских систем. Принцип работы системы показан на рисунке 1.

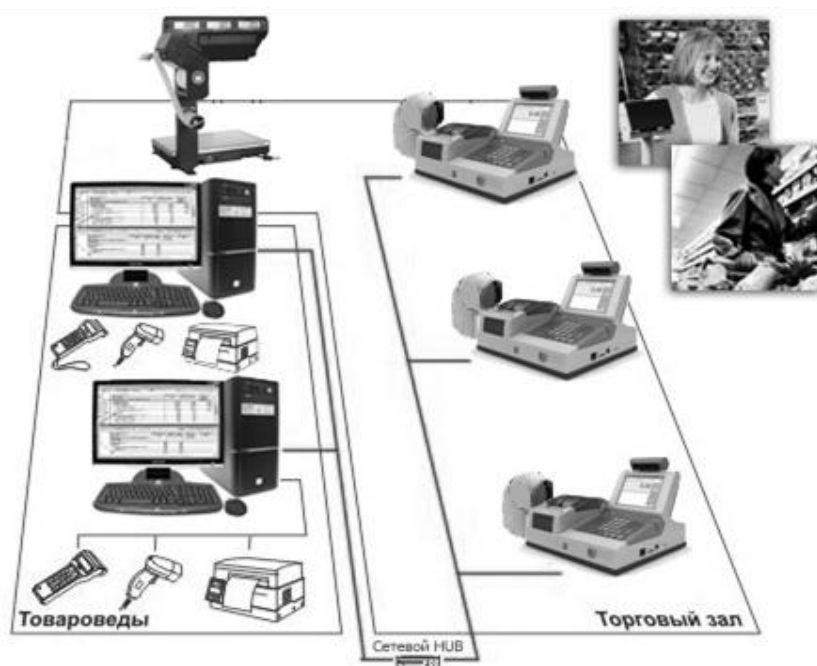


Рисунок 1 – Принцип работы системы

Данная схема состоит из двух отдельных подразделений: товароведов и торгового зала. Рабочее место товароведа оснащено персональным компьютером под управлением системы 1С Предприятие. В качестве вспомогательных устройств выступает терминал сбора данных, принтер этикеток и сканер штрих-кодов. Также рабочее место товароведа может быть оснащено торговыми весами.

Система удобна для работы, так как не требует от пользователя углубленных знаний компьютера, и максимально автоматизирована.

При использовании этой системы автоматизации повышается производительность труда, сокращается время на выписку накладных, сокращается время обработку на данных.

### Литература

1. Байдаков, В. 1С:Предприятие 7.7 Описание встроенного языка / В. Байдаков – Фирма «1С», 2004. – 168 с.

2. Серебренников, Г.Г. 1С:Предприятие 7.7 простые примеры разработки: учебник / Г.Г. Серебренников. – М.: Фирма «1С», 2002. – 80 с.

3. Карпов, Э.А. Организация торговли и маркетинга: учебник / Э.А. Карпов – М.: Старый Оскол, 2010. – 344 с.

**А.В. Сюськин (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Кулинченко**, ст. преподаватель

## **НЕОБХОДИМОСТЬ АВТОМАТИЗАЦИИ ТОРГОВОГО ОБЪЕКТА**

В настоящее время невозможно представить торговый объект без соответствующего оборудования, предназначенного для автоматизации учета товаров. В его состав входят кассовые модули, фискальные регистраторы, считывающее и прикладное оборудование, которое можно встретить, делая покупки в любом магазине. Слабое применение средств оргтехники приводит к снижению производительности труда и эффективности работы управленческого и технического персонала. Степень автоматизации рабочего места торгового объекта влияет на качество обслуживания покупателя, снижение ошибок в работе персонала, скорость обработки данных.

Автоматизация торгового объекта – это комплекс мероприятий по разработке и внедрению в торговый бизнес – процесс высокотехнологичного оборудования и программного обеспечения с целью повышения эффективности использования трудовых ресурсов и качество обслуживания. Кроме того, автоматизация торговли это экономия и увеличение рабочего времени за счет сокращения числа ревизий, возможность прослеживания всего пути товара и денежных средств, контроль остатка товара.

Выделяют аппаратную и программную составляющую автоматизации торгового объекта. К аппаратной части относят: кассовые POS-системы, оборудование штрих-кодирования, принтеры и сканеры штрих-кода, торговые весы с принтером этикеток, терминалы сбора данных, информационные киоски, электронные ценники. К программному обеспечению можно отнести: кассовые программы для управления POS-системой, кассовые средства для управления кассовым узлом, системы управления торговым залом, системы управления предприятием, дополнительные модули управления и программы для автоматизации магазинов.

При использовании данной системы повышается производительность труда, сокращается время на выписку накладных, обработку данных. Сокращается время обрабатываемой информации.

Эффективность работы системы была доказана на проведении соответствующих тестов. Таким образом, данная система удовлетворяет всем требованиям автоматизации торгового объекта.

**Н.Г. Тарачешников (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Леванцов**, ст. преподаватель

## **АВТОМАТИЗАЦИЯ РАСЧЕТА НАЛОГОВЫХ СРЕДСТВ В СРЕДЕ 1С:ПРЕДПРИЯТИЕ 8.1**

Автоматизация позволяет повысить производительность труда, улучшить качество продукции, оптимизировать процессы управления, отстранить человека от производств, опасных для здоровья.

Однако присутствие в решаемых задачах эвристических или сложно программируемых процедур объясняет широкое распространение автоматизированных систем (также, в зависимости от терминологии некоторых исследований, – полуавтоматических систем). Здесь человек участвует в процессе решения, например, управляя им, вводя промежуточные данные. В таких случаях принципиально экономят на защите от редких и сложных нестандартных событий, отводя её роль человеку.

Основная тенденция развития систем автоматизации идет в направлении создания автоматических систем, которые способны выполнять заданные функции или процедуры без участия человека. Роль человека заключается в подготовке исходных данных, выборе алгоритма (метода решения) и анализе полученных результатов. Также в подобных системах предусматривается постепенно наращиваемая защита от нестандартных событий или способы их обхода.

На степень автоматизации влияют вероятность и разнообразность нестандартных событий, продолжительность времени, отведенного на решение задачи, и её вид – типовая или нет.

Основная проблема, с которой сталкиваются сотрудники отдела кадров – это большая трудоемкость управления, огромное количество задач, функций, процессов, которыми необходимо оперативно и качественно управлять.

Все вышеперечисленные проблемы практически невозможно решить без применения современных средств сбора и обработки информации.

Именно с использованием современных программных комплексов достигаются высокие результаты в скорости получения информации и удобства работы с ней. Весь этот процесс перехода от старых принципов работы к информационным технологиям называется автоматизацией.

Автоматизация расчета налоговых средств это тяжелый и трудоемкий процесс, в котором каждая ошибка повлечет за собой серьезные проблемы. Автоматизация исключает ошибки в промежуточных вычислениях пользователем, экономит время, что позволяет задействовать человеческий ресурс на другие виды работ.

Вся автоматизация строится на сложных вычислениях и проверках. Нужно проверить какие данные будут попадать под налогооблагаемую базу, в какую группу входит физическое лицо, и какой процент налогообложения требуется отнять от данной суммы. Все это является сложным механизмом, позволяющим экономить много времени и ресурсов.

Автоматизация процессов расчета налоговых средств позволяет руководству:

1. Снизить временные затраты на выполнение данных функций и высвободить рабочее время службы персонала для решения других задач.
2. Обеспечить достоверность данных.
3. Своевременно получать необходимую информацию по движению и изменению денежных средств, для анализа и принятия управленческих решений.

### Литература

1. Радченко М.Г. 1С:Предприятие 8.2. Практическое пособие разработчика. Примеры и типовые приемы / М.Г.Радченко, Е.Ю. Хрусталева. – М.: ООО «1С-Публишинг», 2009. – 874 с. : ил.
2. Кондраков Н. П. Бухгалтерский учет : рек. М-вом образования РФ / Н. П. Кондраков. – М. : ИНФРА-М, 2008. – 716 с.
3. Селищев, Н.В. 1С:Бухгалтерия предприятия 8.2. Практическое пособие. – Кнорус, 2014. – 392 с.
4. Хрусталева, Е.Ю. Разработка сложных отчетов в 1С:Предприятие 8. Система компоновки данных / Е.Ю. Хрусталева. – 2-е изд. – М.: ООО «1С-Публишинг», 2012. – 484 с.
5. Филатова, В.О. 1С для начинающих. Понятный самоучитель. – СПб.: Питер, 2013. – 256 с.
6. Габец, А.П. Реализация прикладных задач в системе «1С:Предприятие 8.2» / А.П. Габец, Д.В. Козырев, Д.С. Кухлевский, Е.Ю. Хрусталёва. – М.: ООО «1С-Публишинг», 2012. – 714 с.



**Н.Г. Тарачешников (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **В.Н. Леванцов**, ст. преподаватель

## **АВТОМАТИЗАЦИЯ ПРОВЕДЕНИЯ НАЧИСЛЕНИЙ РАБОТНИКАМ ОРГАНИЗАЦИИ В СРЕДЕ 1С:ПРЕДПРИЯТИЕ 8.2**

Автоматизация позволяет повысить производительность труда, улучшить качество продукции, оптимизировать процессы управления, отстранить человека от производств, опасных для здоровья.

Однако присутствие в решаемых задачах эвристических или сложно программируемых процедур объясняет широкое распространение автоматизированных систем (также, в зависимости от терминологии некоторых исследований, – полуавтоматических систем). Здесь человек участвует в процессе решения, например, управляя им, вводя промежуточные данные. В таких случаях принципиально экономят на защите от редких и сложных нестандартных событий, отводя её роль человеку.

Широкое внедрение АСУ – это объективная необходимость, обусловленная усложнением задач управления, повышением объемов информации, которые необходимо перерабатывать в системе управления энергосистемами и предприятиями.

Основная цель автоматизации и компьютеризации управления – обеспечение оптимального функционирования объекта управления путем правильного выбора целей и средств, для их достижения с учетом имеющихся ограничений, наилучшего распределения заданий между отдельными частями, из которых состоит объект.

Основной экономической эффект от внедрения АСУ учреждений получается за счет повышения уровня планирования, лучшей организации производства, более полной загрузки оборудования, обеспечения ритмичности работы учреждения, сокращения непроводительных потерь, что в итоге повышает производительность труда и снижает издержки производства.

В современном мире самые разнообразные компании, предприятия, склады и магазины всё чаще используют автоматизированные системы ведения и учёта данных. Говоря более прямо, ни одна компания, не использующая подобную систему, не сможет составить конкуренцию в эффективности производства компании, использующей систему автоматизированного учёта. Ведь такая система, существенно упрощает и при этом ускоряет процесс обработки и учёта информации.

На степень автоматизации влияют вероятность и разнообразность нестандартных событий, продолжительность времени, отведенного на решение задачи, и её вид – типовая или нет.

Автоматизация расчета налоговых средств это тяжелый и трудоемкий процесс, в котором каждая ошибка повлечет за собой серьезные проблемы. Автоматизация исключает ошибки в промежуточных вычислениях пользователем, экономит время, что позволяет задействовать человеческий ресурс на другие виды работ.

Основная проблема, с которой сталкиваются сотрудники отдела кадров – это большая трудоемкость управления, огромное количество задач, функций, процессов, которыми необходимо оперативно и качественно управлять.

Все вышеперечисленные проблемы практически невозможно решить без применения современных средств сбора и обработки информации. Именно с использованием современных программных комплексов достигаются высокие результаты в скорости получения информации и удобства работы с ней. Весь этот процесс перехода от старых принципов работы к информационным технологиям называется автоматизацией.

Вся автоматизация строится на сложных вычислениях и проверках. Нужно проверить какие данные будут попадать под налогооблагаемую базу, в какую группу входит физическое лицо, и какой процент налогообложения требуется отнять от данной суммы. Все это является сложным механизмом, позволяющим экономить много времени и ресурсов.

Автоматизация процессов расчета налоговых средств позволяет руководству:

1. Снизить временные затраты на выполнение данных функций и высвободить рабочее время службы персонала для решения других задач.

2. Обеспечить достоверность данных.

3. Своевременно получать необходимую информацию по движению и изменению денежных средств, для анализа и принятия управленческих решений.

Вся деятельность организации напрямую зависит от того, какой квалификацией обладают его сотрудники, насколько эта квалификация соответствует задачам организации, поддерживаются ли данные задачи соответствующей системой мотивации, как те или иные управленческие решения влияют на эффективность работы персонала. Таким образом, можно утверждать, что человеческий капитал является одним из ключевых ресурсов управления.

Автоматизация налоговых средств является очень важным звеном в бухгалтерском учете. Налоговые средства на предприятии рассчитываются как процент от начисления. Процент может быть как фиксированный, так и динамический. При формировании документа выполняется автоматический расчет налогов. Так же существует налогооблагаемая база, от которой и считается процент. Для каждой организации высчитывается, в соответствии с законодательством предел, при котором подоходный налог не взимается.

При исчислении НДС является выручка от реализации товаров, работ, услуг, определяемая, исходя из всех доходов в денежной и натуральной форме.

Выручка в иностранной валюте пересчитывается в валюту РБ на дату фактической реализации товаров, работ, услуг.

При реализации товаров по ценам ниже остаточной стоимости, по ценам ниже приобретения для товаров налогооблагаемая база определяется исходя из остаточной стоимости и цены приобретения.

Предел рассматривается в разрезе месяца, т. е. каждый месяц предел записывается как новый и вычитается от общих начислений за месяц. Для подсчета облагаемой базы необходимо создать промежуточные итоги, т. к. если сумма общих начислений превышает предел, тогда каждая последующая единица начисления облагается налогом. Для обычных организаций предел формируется в зависимость от размера начисления, в то время как для иных организаций существует фиксированный предел.

## Литература

1. Радченко, М.Г. 1С:Предприятие 8.2. Практическое пособие разработчика. Примеры и типовые приемы / М.Г. Радченко, Е.Ю. Хрусталева. – М.: ООО «1С-Публишинг», 2009. – 874 с. : ил.

2. Кондраков, Н. П. Бухгалтерский учет : рек. М-вом образования РФ / Н. П. Кондраков. – М. : ИНФРА-М, 2008. – 716 с.

3. Селищев, Н.В. 1С:Бухгалтерия предприятия 8.2. Практическое пособие. – Кнорус, 2014. – 392 с.

4. Хрусталева, Е.Ю. Разработка сложных отчетов в 1С:Предприятие 8. Система компоновки данных / Е.Ю. Хрусталева. – 2-е изд. – М.: ООО «1С-Публишинг», 2012. – 484 с.

5. Габец, А.П. Реализация прикладных задач в системе «1С:Предприятие 8.2» / А.П. Габец, Д.В. Козырев, Д.С. Кухлевский, Е.Ю. Хрусталева. – М.: ООО «1С-Публишинг», 2012. – 714 с.

**О.М. Ткачев (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **Е.А. Ружицкая**, канд. физ.-мат. наук, доцент

## **РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ «ВИКТОРИНА» ДЛЯ ПЛАТФОРМЫ IOS**

Сегодня множество людей пользуются мобильными устройствами от Apple – iPhone, iPad и т. д. Актуальным направлением является создание приложений для мобильных платформ. Поэтому было разработано мобильное приложение «Викторина» для платформы iOS.

Приложение «Викторина» позволяет оценить уровень фундаментальных знаний пользователя по общеобразовательным предметам. Вопросы викторины охватывают различные темы из разных областей науки. Данное приложение дает возможность играть в соревновательном режиме. Все вопросы разбиты на несколько категорий. В конце выводится в процентах количество заработанных очков каждым пользователем. После чего, есть возможность начать викторину заново или же покинуть ее. Отвечая на вопросы, Вы зарабатываете очки и поднимаетесь в общем рейтинге. На каждый вопрос можно получить подсказку. За каждую подсказку количество заработанных очков уменьшается.

Возможности приложения: большая база вопросов; три уровня сложности для прохождения викторины; возможность использования подсказок в каждом уровне; подробная статистика; рейтинг.

Для написания приложения использовались следующие технологии: Swift, Objective-C, iOS SDK, UIKit. SDK – комплект средств разработки, который позволяет создавать приложения для определённого пакета программ, программного обеспечения базовых средств разработки, аппаратной платформы, компьютерной системы, видеоигровых консолей, операционных систем и прочих платформ. SDK обладает возможностями, недоступными для web-технологий. Swift – универсальный язык программирования, созданный для операционных систем OS X и iOS. Он основан на базе C и Objective-C, но при этом включает в себя все последние технологии, разработанные инженерами Apple в течение последних лет (ARC и LLVM). Swift позиционируется как современный, быстрый, интерактивный и безопасный язык программирования. UIKit – современный, быстрый и модульный CSS-и js-фреймворк, содержащий в себе большой набор компонент.

Разработка данного приложения осуществлялась в среде разработки Xcode. Создание графического интерфейса происходило в Interface Builder. В процессе разработки была использована модель данных MVC.

Разработанное приложение можно позиционировать как законченный продукт, готовый к использованию на мобильной платформе iOS.

**В.С. Тотиков (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **Е.Е. Пугачёва**, ассистент

## **АВТОМАТИЗАЦИЯ ПРОИЗВОДСТВЕННОЙ ДЕЯТЕЛЬНОСТИ СКЛАДА АВТОЗАПЧАСТЕЙ ООО «ШИМ-БИ ТРАК» В СИСТЕМЕ ПРОГРАММ 1С:ПРЕДПРИЯТИЕ 8.**

Для автоматизации деятельности склада автозапчастей ООО «ШиМ-Би Трак» были созданы следующие объекты конфигураций.

Объект конфигурации *Справочник* предназначен для работы со списками данных. Как правило, в работе любой фирмы используются списки сотрудников, списки товаров, списки клиентов, поставщиков и т. д.

Объект конфигурации *Документ* предназначен для описания информации о совершенных хозяйственных операциях или о событиях, произошедших в жизни организации вообще.

Объект конфигурации *Перечисление* предназначен для описания структуры хранения постоянных наборов значений, не изменяемых в процессе работы конфигурации. На основе объекта конфигурации *Перечисление* платформа создает в базе данных таблицу, в которой может храниться набор некоторых постоянных значений.

Объект конфигурации *Регистр сведений* предназначен для описания структуры хранения данных в разрезе нескольких измерений. На основе объекта конфигурации *Регистр сведений* платформа создает в базе данных таблицу, в которой может храниться произвольная информация, «привязанная» к набору измерений.

Объект *Регистр накопления* предназначен для описания структуры накопления данных. На основе объекта конфигурации *Регистр накопления* платформа создает в базе данных таблицы, в которых будут накапливаться данные, «поставляемые» различными объектами базы данных.

Объект конфигурации *Отчет* предназначен для описания алгоритмов, при помощи которых пользователь сможет получать необходимые ему выходные данные. Алгоритм формирования выходных данных описывается при помощи визуальных средств или с использованием встроенного языка. В реальной жизни объектам конфигурации *Отчет* соответствуют всевозможные таблицы выходных данных, сводных данных, диаграммы.

Полученная база данных легка и понятна в использовании, дает всю необходимую информацию пользователю, а также обеспечивает быстрый и легкий доступ к данным. Разработанная подсистема дает возможность обрабатывать большие объемы информации, имеет удобный интерфейс для работы с данными.

**В.С. Тотиков (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **Е.Е. Пугачёва**, ассистент

## **АВТОМАТИЗАЦИЯ МАГАЗИНА АВТОЗАПЧАСТЕЙ В СИСТЕМЕ ПРОГРАММ 1С:ПРЕДПРИЯТИЕ 8. ОСНОВНЫЕ ВОЗМОЖНОСТИ ДАННОЙ СИСТЕМЫ И ОБЛАСТИ ЕЁ ПРИМЕНЕНИЯ**

1С:Предприятие – это специализированная объектно-ориентированная система управления базами данных (СУБД), предназначенная для автоматизации деятельности предприятия. Ориентируется на автоматизации учетных задач: кадровый учет, расчет зарплаты, бухгалтерский учет, складской учет и т. д.

Система программ «1С:Предприятие» включает в себя платформу и прикладные решения, разработанные на ее основе, для автоматизации деятельности организаций и частных лиц. Сама платформа не является программным продуктом для использования конечными пользователями, которые обычно работают с одним из многих прикладных решений (конфигураций), разработанных на данной платформе. Такой подход позволяет автоматизировать различные виды деятельности, используя единую технологическую платформу.

Гибкость платформы позволяет применять 1С:Предприятие в самых разнообразных областях:

- автоматизация производственных и торговых предприятий, бюджетных и финансовых организаций, предприятий сферы обслуживания и т. д.;

- поддержка оперативного управления предприятием;

- автоматизация организационной и хозяйственной деятельности;

- ведение бухгалтерского учета с несколькими планами счетов и произвольными измерениями учета, регламентированная отчетность;

- широкие возможности для управленческого учета и построения аналитической отчетности, поддержка многовалютного учета;

- решение задач планирования, бюджетирования и финансового анализа;

- расчет зарплаты и управление персоналом;

- другие области применения.

Платформа «1С:Предприятие» представляет собой программную оболочку над базой данных. Кроме того, с версии 8.1 хранение данных возможно в СУБД PostgreSQL и IBM DB2, а с версии 8.2 добавилась и Oracle. Имеет свой внутренний язык программирования, обеспечивающий, помимо доступа к данным, возможность взаимодействия

с другими программами посредством OLE и DDE. Клиентская часть платформы функционирует только в среде ОС Microsoft Windows. Начиная с версии 8.1, серверная часть платформы в клиент-серверном варианте работы «1С:Предприятия» может функционировать на ОС Linux.

Существуют следующие версии платформы:

- учебная – допускает конфигурирование, имеет весьма существенные ограничения;
- базовая – допускает использование только Базовых конфигураций, не допускает конфигурирования, не допускает смешивания «компонент»;
- стандартная – только «Бухгалтерский учёт», допускает конфигурирование, имеет ограничения;
- профессиональная – обладает максимальными возможностями среди однопользовательских версий, допускает совместное использование «компонент»;
- сетевая – совместное использование в локальной сети с ограничением числа пользователей или без ограничений:

SQL – обладает максимальными возможностями, допускает хранение данных в MS SQL Server v.6, v.7 или 2000. Также существуют неофициальные патчи bkend.dll, позволяющие использовать MS SQL 2005 и 2008 (с созданием представления для базы данных через SQL Server Management Studio для последнего).

Логически всю систему можно разделить на две большие части, которые тесно взаимодействуют друг с другом: конфигурацию и платформу, которая управляет работой конфигурации. Основной особенностью системы 1С:Предприятия является ее конфигурируемость.

Собственно система 1С:Предприятие представляет собой совокупность механизмов, предназначенных для манипулирования различными типами объектов предметной области. Конкретный набор объектов, структуры информационных массивов, алгоритмы обработки информации определяет конкретная конфигурация.

Программа «1С:Предприятие 8. Магазин автозапчастей» разработана для комплексной автоматизации магазинов розничной торговли специализирующихся на продаже товаров автомобильной тематики. От автозапчастей, шин и дисков, до масел, автокрасок и сопутствующих товаров. Данное отраслевое решение станет незаменимым в работе, как одиночного розничного магазина, так и розничных торговых сетей. Круг потенциальных пользователей «1С:Предприятие 8.

Магазин автозапчастей» очень широк: от консультантов, продавцов и кассиров до директоров, руководителей и маркетологов. Программа позволяет автоматизировать учет товарных запасов на складах магазинов и учет денежных средств в кассах организаций.

«1С:Предприятие 8. Магазин автозапчастей» автоматизирует учет всех операций, производящихся в магазине:

- заказ товаров поставщику;
- поступление товаров от контрагента на склад магазина;
- перемещение товаров между складами магазина;
- передача товаров между организациями;
- возврат товаров поставщику;
- возврат товаров от покупателей;
- продажа комплектов товаров;
- пересчет товаров (инвентаризация);
- реализация товаров и услуг;
- заказ товаров от покупателя;
- поступление денежных средств от покупателей;
- перемещение денежных средств между кассами магазина;
- оформление чеков продажи и сводного отчета по контрольно-кассовой машине;
- работа с эквайринговыми системами и с банковскими кредитами.

В программе «1С:Предприятие 8. Магазин автозапчастей» реализованы возможности:

- оформление прихода товаров от контрагента на склады магазина, в том числе в двухфазовом режиме;
- оформление перемещения товаров между магазинами, внутренними складами магазинов, магазинами и складами предприятия, в том числе в двухфазовом режиме;
- взаимодействие с автомобильными каталогами предоставляет возможность поиска деталей в базе данных каталога и импорта данных из каталога в номенклатурный справочник конфигурации;
- указание аналогов и применяемости автозапчастей к различным маркам автомобилей.

Таким образом, с помощью «1С: Предприятие 8. Магазин автозапчастей» решается весь контур учетных задач розничного магазина: управление складскими запасами, учет движения товара между магазинами, контроль взаиморасчетов с поставщиками, управление маркетинговыми мероприятиями (скидки, наценки), регистрация розничных продаж.



**Г.А. Усольцев («ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **Д.С. Кузьменков**, канд. физ.-мат. наук, доцент

## **СОЗДАНИЕ ВЕБ-ПРИЛОЖЕНИЙ НА ПЛАТФОРМЕ GRAILS**

Grails является программной платформой для быстрой и простой разработки веб-приложений. Этот фреймворк разработан с помощью языка Groovy, который в свою очередь основан на Java. Grails был создан под сильным влиянием Ruby on Rails, основывается на шаблоне «модель-представление-контроллер» (MVC).

Данный фреймворк ставит своей задачей предоставить разработчикам максимальные возможности для разработки приложений с использованием базовой технологии и многочисленных плагинов, привносящих в платформу дополнительный функционал. Среди ключевых возможностей доступных без подключения дополнительных плагинов есть простой в использовании механизм ORM, основанный на Hibernate, технология представлений (GSP), уровень контроллеров, построенный на Spring MVC, встроенный Tomcat сервер с возможностью перезагрузки «налету» для ускорения разработки, поддержка интернационализации, транзакционный уровень сервисов и др.

Платформа Grails обладает рядом характеристик, позволяющих ускорить процесс разработки приложений:

- отсутствие XML конфигураций;
- готовая к использованию среда разработки, также расширяемое плагинами;
- функциональность, доступная благодаря использованию примесей (mixins).

Grails добавляет некоторые методы ряда классов, используя примеси. Примесь – метод, который добавляется классу динамически. Эти динамические методы позволяют разработчикам выполнять операции без необходимости реализовывать интерфейсы или расширять базовые классы. Grails предоставляет динамические методы в зависимости от типа класса. Например, доменные классы содержат методы автоматизации операций персистенции.

Также, значительного ускорения процесса разработки позволяет добиться командная строка Grails, разработанная на основе инструмента сборки Gant. Командная строка позволяет создавать основы для реализации различных элементов приложения, к примеру контроллеры, доменные классы, представления фильтры, сервисы и т. д., используя всего лишь несколько команд. Стоит отметить, что при создании классов, используемых в приложении, автоматически создаются заготовки для юнит-тестов для этих классов.

Автоматическое тестирование – одна из ключевых возможностей фреймворка. Поэтому Grails предоставляет множество различных возможностей для облегчения процесса создания тестов. Тесты в Grails разделяются на три вида: юнит-тесты, интеграционные тесты и функциональные тесты, и отличаются возможностями, предоставляемыми платформой во время выполнения тестов.

Grails не только является гибким фреймворком с большим количеством возможностей, доступных «из коробки», но также значительно расширяем с помощью плагинов. Плагины могут предоставлять различный функционал, включая поддержку различных библиотек кэширования, безопасности, поддержку нереляционных баз данных для GORM (ORM фреймворк для Grails), а также различных front-end библиотек и фреймворков. Grails основан на языке Groovy – объектно-ориентированном языке программирования, разработанный для платформы Java. Groovy имеет значительное количество возможностей, отличающих его от Java, в том числе:

- статическая и динамическая типизация;
- встроенный синтаксис для списков, ассоциативных массивов, регулярных выражений и т. д.;
- замыкания;
- перегрузка операций.

Синтаксис Groovy является Java-подобным (до такой степени, что большинство кода на Java будет являться верным кодом на Groovy) с динамической компиляцией в JVM байт-код и поэтому может работать с другими проектами (например, библиотеками), написанными на Java или другом языке на основе JVM (например, Scala). Одной из важных возможностей Groovy является создание и использование предметно-ориентированных языков (domain-specific languages). Данная возможность основывается на ряде ключевых возможностей Groovy, в том числе на использовании «командных цепочек» (последовательный вызов методов без использования скобок для передачи аргументов), а также на перегрузке операторов, и позволяет реализовывать и поддерживать некоторые операции над доменными объектами.

Было реализовано приложение в среде Grails, позволяющее организовать удаленное централизованное хранение файлов пользователя и работу с ними.

## Литература

1. The Grails Framework – Reference Documentation [Электронный ресурс] – URL: <http://grails.github.io/grails-doc/latest/> (дата обращения 27.01.2015).

**Т.В. Филонюк (УО «БТЭУ» г. Гомель)**  
Науч. рук. **С.В. Карпенко**, канд. физ.-мат. наук, доцент

## **АВТОМАТИЗАЦИЯ В СКЛАДСКОЙ ЛОГИСТИКЕ – ВАЖНОЕ КОНКУРЕНТНОЕ ПРЕИМУЩЕСТВО**

Складская логистика – это специально организованная совокупность связанных частей, помогающая найти рациональное размещение материального потока на складе и оптимальное управление им.

Одной из основных функциональных областей логистики является транспортировка продукции. Иначе говоря, потребителю нужен качественный товар в нужном количестве, в нужном месте, в нужное время и доставленный с минимальными затратами. Все перечисленные выше моменты указывают на значимость критерия логистической системы «точно в срок».

Данный критерий предполагает оптимальный выбор вида транспорта и составление графика обслуживания потребителей, что позволяет успешно выполнять задачи снабжения точно в срок.

Задачи, решаемые логистической системой, и выработку по ним стратегии можно разделить на три группы:

1. Задачи, связанные с формированием рыночных зон обслуживания, прогнозом материалопотока, его обработкой в обслуживающей системе (склад поставщика/потребителя, предприятия оптовой торговли) и другими работами по оперативному управлению и регулированию материального потока.

2. Задачи, включающие разработку системы организации транспортного процесса (план перевозок, план распределения вида деятельности, план формирования грузопотоков, график движения транспортных средств и др.).

3. Задачи, связанные с управлением запасами на организациях, складских комплексах, размещение запасов и их обслуживание транспортными средствами, информационными системами.

Перемещение материальных потоков в логистической цепи невозможно без концентрации в определенных местах необходимых запасов, для хранения которых предназначены соответствующие склады. Движение через склад связано с затратами живого и овеществленного труда, что увеличивает стоимость товара. В связи с этим проблемы, связанные с функционированием складов, оказывают значительное влияние на рационализацию движения материальных потоков в логистической цепи, использование транспортных средств и издержек обращения.

Рассмотрим процесс «Деятельность склада». Рассмотрим основные работы процесса проектирования.

Весь процесс деятельности склада подразделяется на:

- Приемку товара (принятие товара по сопроводительным документам и передача его на хранение). На данном этапе товар с сопроводительными документами поступает на склад и подвергается подсчету. Но, каким бы ни был результат подсчета товара (положительным либо отрицательным), он после этого вместе с сопроводительными документами передается на хранение.

- Отгрузку и возврат товара (выдача скомплектованного товара клиенту, либо возврат поставщику). Данный этап подразумевает отгрузку клиенту товара, скомплектованного по отгрузочным документам.

- Хранение (основная и самая сложная функция склада, подразумевает все остальные действия с товаром, например, складирование, комплектование, оформление документации на товар, списание). Процесс хранения, в свою очередь подразделяется на:

- формирование отгрузочных документов;
- складирование;
- комплектование;
- списание товара.

Диаграмма складирования, в свою очередь подразделяется на:

- складирование на оптимальный склад (при поступлении товара, принятого в соответствии с приходными документами).

- складирование на возвратный склад (при поступлении товара, не принятого в соответствии с приходными документами).

- формирование возвратных документов (формирование и передача документов на комплектование товара для возврата поставщику).

Проиллюстрируем модель процесса «Деятельность склада» (рисунок 1).

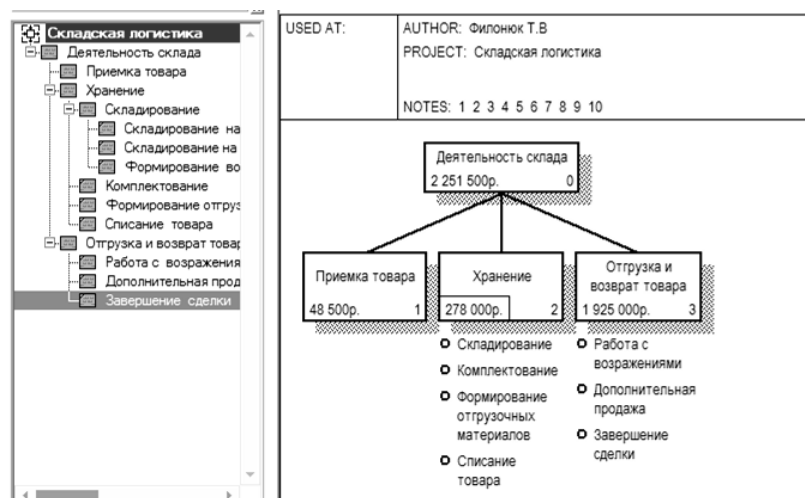


Рисунок 1 – Модель процесса «Деятельность склада»

Для проведения функционально-стоимостного анализа следует учитывать стоимостное выражение функций. Для этого объединяют матрицы сравнения функций и затраты на выполнение каждой функции в единую таблицу. В основе анализа лежат данные, которые обеспечивают менеджеров информацией, необходимой для обоснования и принятия управленческих решений.

Проиллюстрируем функционально-стоимостной анализ модели AS-IS (рисунок 2).

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Соотношение затрат и относительных значимостей недекомпозированных функций, %												
2		Функции											
3	Показатели	A21.1	A21.2	A21.3	A22	A23	A24	A1	A31	A32	A33		
4	Затраты	770 000	535 000	1 360 000	535 000	535 000	265 000	48 500	860 000	625 000	1 460 000		6 993 500
5	Относительная значимость	0	107 000	272 000	107 000	107 000	53 000	14 550	0	31 250	73 000		764 800
6													
7													
8		Функции											
9	Показатели	A21.1	A21.2	A21.3	A22	A23	A24	A1	A31	A32	A33		
10	Затраты	11	8	19	8	8	4	1	12	9	21		100
11	Относительная значимость	0	13,99	35,56	13,99059	13,99	6,93	1,90	0,00	4,09	9,54		100
12													

Рисунок 2 – Функционально-стоимостной анализ модели AS-IS

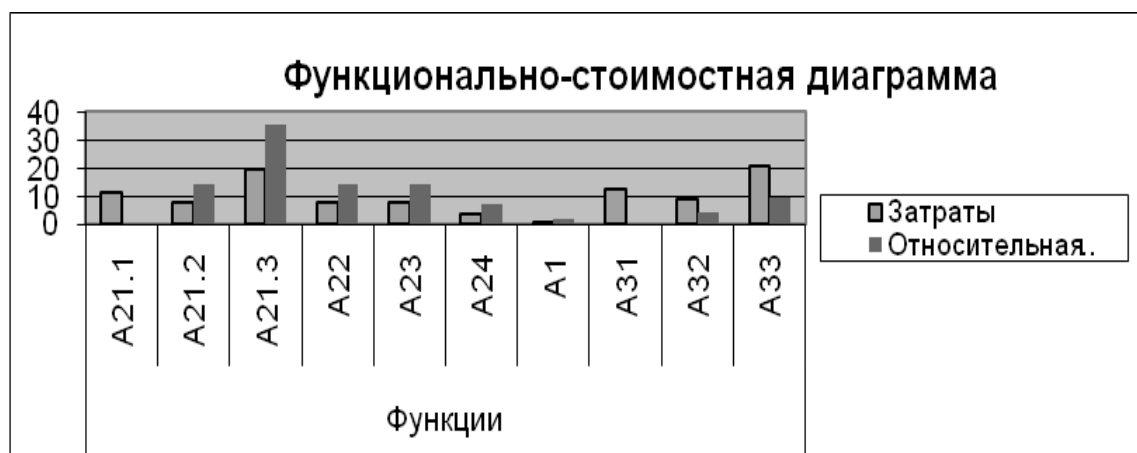


Рисунок 3 – Функционально-стоимостная диаграмма модели AS-IS

Автоматизируем деятельность склада информационными технологиями. Для этого ручную работу, которую выполнял персонал склада, переведем на машинную обработку. Для уменьшения стоимости всего БП уменьшим затраты по наиболее затратным функциям. Если процедуры, такие как Работа с возражениями, Дополнительная продажа, будут выполняться быстрее, по сравнению с текущим ведением дел, то это позволит сократить издержки этих функций, а следовательно и всего БП.

## Литература

1. Гаджинский, А.М. Логистика / А.М. Гаджинский. – М.: Дашков и К, 2006.
2. Интегрированная логистика накопительно-распределительных комплексов (склады, транспортные узлы, терминалы: учеб. для транспортных вузов / под общ. ред. Л.Б. Миротина. – М.: Экзамен, 2003. – 448 с.
3. Логистика: учебник / под ред. Б.Н. Аникина. – 3-е изд., перераб. и доп. – М.: ИНФРА-М, 2001 – 2004. – 352 с.

**А.В. Філіпчык (УА БДУІР, Мінск)**

Нав. кір. **Сярэбраная Л.В.**, канд. тэхн. навук, дацэнт

### **АЛГАРЫТМ ТРАНСКРЫБАВАННЯ ЛЕКСІЧНЫХ АДЗІНАК У ЛІНГВІСТЫЧНАЙ ІНФАРМАЦЫЙНА-ПОШУКАВАЙ СІСТЭМЕ**

Разгледжаныя асаблівасці транскрыбавання ў лінгвістычных інфармацыйна-пошукавых сістэмах. Прапанаваны алгарытм, які падыходзіць для рашэння пастаўленай задачы транскрыбавання лексічных адзінак.

Форма захоўвання дадзеных залежыць ад спецыфікі саміх дадзеных, а таксама ад спецыфікі іх прымянення. У некаторых выпадках трэба ведаць не толькі правільнае напісанне слова, але яшчэ і правільнае яго вымаўленне. Напрыклад, пры выхадзе на рынкі замежных краін можа аказацца, што гучанне назвы пэўнага брэнда прымае зусім іншы кантэкст (так, французскі аўтамабіль Chevrolet Nova пацярпеў фіяска ў Іспаніі, бо “No va” па-іспанску азначае “не рухаецца”). Як прыклад рэальнай сістэмы, якая працуе з транскрыпцыямі, можна прывесці камерцыйны прадукт Soundex, што прымяняецца авіякампаніямі для захоўвання імён і прозвішчаў пасажыраў у фанетычнай форме для прадухілення канфліктных сітуацый з няправільным напісаннем ідэнтыфікацыйных дадзеных у розных мовах і алфавітах. Для працы такіх сістэм неабходныя эфектыўныя алгарытмы транскрыбавання, гэтай тэме і прысвечана дадзеная праца.

Распрацоўка алгарытма вымагае добрага ведання правілаў вымаўлення канкрэтнай мовы і кадавання яе на пісьме. У агульным выпадку транскрыпцыя кожнай літары можа залежаць не толькі ад яе самой, але і ад папярэдніх і наступных літар, націску, ад становішча

літары ў слове (у пачатку слова, у сярэдзіне ці ў канцы), а таксама ад лексічнага кантэксту (калі транскрыбуецца не асобнае слова, а тэкст). Правілы транскрыбавання спецыфічныя для кожнай мовы, таму ў дадзенай працы была абрана беларуская мова.

На ўваход распрацаванага алгарытма транскрыпцыі могуць паступаць як асобныя словы, так і цэлыя тэксты. На выхадзе мы атрымліваем транскрыпцыю – набор сімвалаў, які максімальна блізка перадае асаблівасці вымаўлення ўваходнага слова (тэксту).

Алгарытм транскрыпцыі можна падзяліць на наступныя этапы:

1. Марфалагічны аналіз для выяўлення патэнцыйных аманімічных канфліктаў.
2. Вызначэнне націскага склада ў кожным слове.
3. Пафанемная транскрыпцыя з улікам лексічнага кантэксту.

На першым этапе алгарытм праводзіць марфалагічны аналіз кожнай лексічнай адзінкі для выяўлення выпадкаў амаграфіі. Відавочна, што зварот да слоўніка не заўсёды дазваляе адназначна вызначыць прыналежнасць славаформы да той ці іншай лексемы, а ад гэтага можа залежыць націск і вымаўленне. Амаграфы – славаформы з аднолькавым напісаннем, якія, тым не менш, належаць да розных лексем і могуць адрознівацца націскам і граматычнымі характарыстыкамі. Метад кантэкстнага аналіза ўлічвае славаформы ў левым і правым кантэксце і падлічвае верагоднасць з'яўлення ў дадзеным кантэксце той ці іншай граматычнай формы. У выпадку, калі граматычныя характарыстыкі амаграфіі супадаюць, прымяняецца аналіз далёкага кантэксту: улічваюцца ўсе іншыя словы ў сказе.

Другі этап – вызначэнне націскага складу. Для працы алгарытм выкарыстоўвае базу славаформ, пабудаваную на аснове слоўнікаў беларускай мовы, узятых з адкрытых крыніц (усяго больш за 500.000 уваходжанняў). Для словаў, што ўваходзяць у гэтыя слоўнікі, націск ужо вызначаны. Аднак застаецца праблема апрацоўкі лексічных адзінак, якія адсутнічаюць у слоўніках: імёны, назвы, рэгіяналізмы, аўтарскія словы, неалагізмы. Алгарытм павінны з высокай імавернасцю правільна вызначаць націск у такіх словах.

У аснове метада вызначэння націска – формула  $p = (n + 1)/2$ , дзе  $n$  – колькасць складаў у слове. Гэты алгарытм дае добрыя вынікі ў кароткіх словах, але ў доўгіх словах, асабліва складаных, дае даволі вялікую колькасць памылак. Для паляпшэння вынікаў выкарыстоўваецца механізм вылучэння стандартных прэфіксаў, такіх як трактара-, стара-, электра- і г. д. Слова дзеліцца на дзве часткі: прэфікс і аснову, якая шукаецца ў слоўніку ці зноў разлічваецца па формуле. Таксама робіцца аналіз на ўваходжанне ў слова стандартных суфіксаў

(напрыклад, звычайна паслянаціскны *-нік*) і стандартных частак спецыфічных прозвішчаў (прозвішчы на *-швілі* і *-адзэ*).

На трэцім этапе адбываецца пафанемная транскрыпцыя. Алгарытм прымяняе для кожнай фанемы пэўны набор правілаў, якія вызначаюць яе правільнае гучанне. Можна адзначыць наступныя групы правілаў: дыфтонгі, аглушэнне і азванчэнне, асіміляцыя, змягчэнне, рэдукцыя. Алгарытм таксама ўлічвае знакі прыпынку, якія вызначаюць паўзы паміж словамі ў тэксце. Наяўнасць знака прыпынку звычайна азначае, што нельга прымяняць правілы кантэксту суседняга слова, бо словы не вымаўляюцца злітна.

Атрыманы алгарытм дазваляе ствараць фанетычныя транскрыпцыі як для асобных словаў, так і для тэкстаў, пасля чаго перадаваць яе далей для вырашэння канкрэтных мэтаў сістэмы. Алгарытм быў распрацаваны ў межах стварэння лінгвістычнай інфармацыйна-пошукавай сістэмы, якая спецыялізуецца на аўтаматызацыі падбору рыфм і напісання вершаваных радкоў. Таксама дадзены алгарытм можа быць выкарыстаны ў сістэмах аўтаматычнага сінтэзу маўлення, аналізатарах галасавых каманд і іншых сферах.

### Літаратура

1. Кипяткова, И.С. Разработка и оценивание модуля транскрибирования для распознавания и синтеза русской речи / И.С. Кипяткова, А.А. Карпов // Научно-теоретический журнал «Искусственный интеллект». – 2009. – № 2 – С. 18–26.

2. Кондратов, А. Математика и поэзия / А. Кондратов. – М.: Знание, 1962. – 42 с.

3. Slounik.org [Электронны рэсурс]: беларускія слоўнікі і энцыклапедыі – электронныя тэкставыя дадзеныя – Рэжым доступу: <http://www.slounik.org>.

**Н.С. Фомин (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **Г.Л. Карасёва**, канд. физ.-мат. наук, доцент

### ИСПОЛЬЗОВАНИЕ СРЕДСТВ MYSQL И PHP ПРИ СОЗДАНИИ БАЗ ДАННЫХ

PHP – это скриптовый язык программирования общего назначения, интенсивно применяемый для разработки веб-приложений. В настоящее время поддерживается подавляющим большинством хостинг-провайдеров и является одним из лидеров среди языков



программирования, применяющихся для создания динамических веб-сайтов.

MySQL – это свободная реляционная система управления базами данных.

Цель работы заключается в разработке и создании базы данных для стоматологической клиники средствами MySQL и PHP.

Для этого была разработана и написана базы данных средствами MySQL:

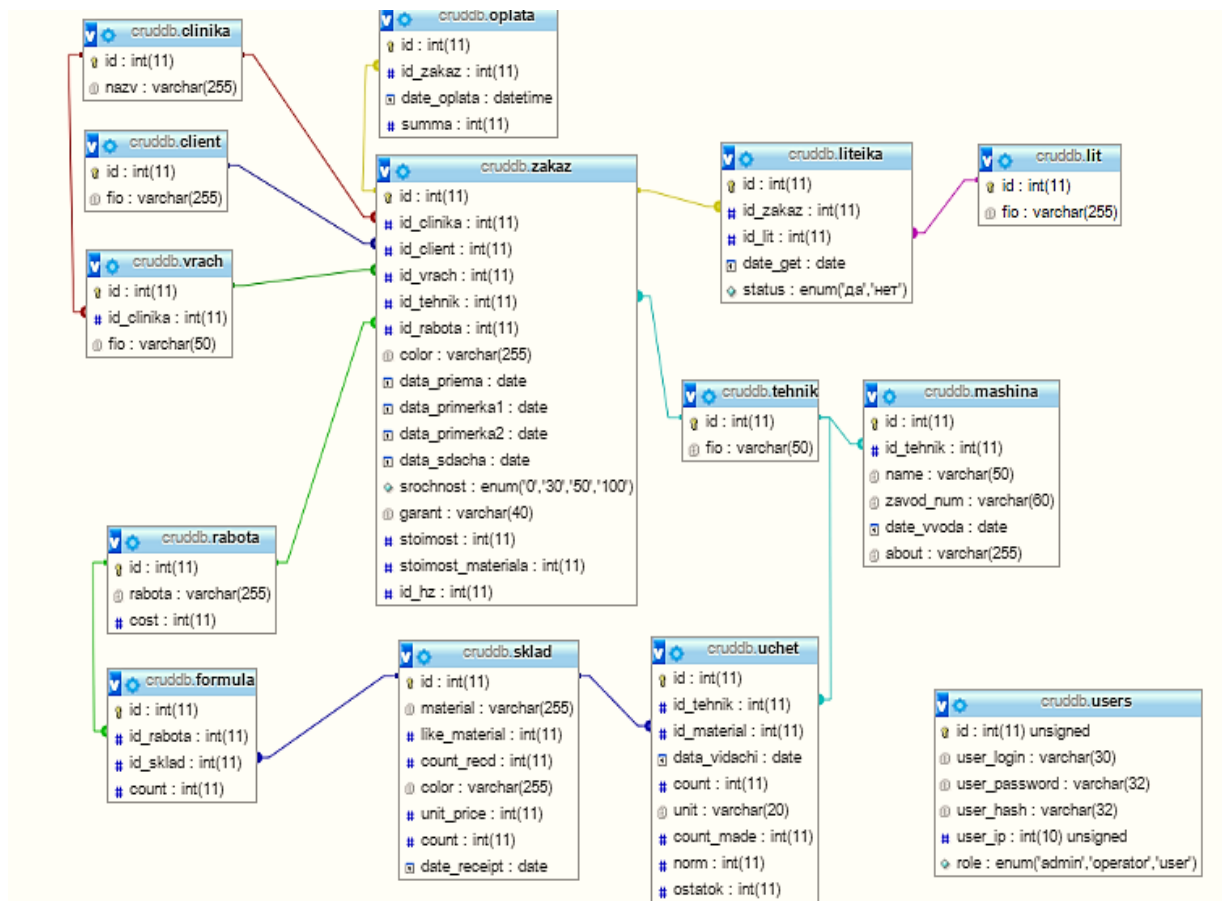


Рисунок 1

Далее для каждой таблицы был описан отдельный класс, а также отдельные страницы для добавления, изменения, удаления и отображения данных.

Подключение к базе данных было создано при помощи средств PHP. Также по заданию было создано сохранение данных в Excel и подключение принтера печати штрих-кодов при помощи сторонних PHP -библиотек.

Внешний вид сайта был создан средствами JavaScript и CSS.

Таким образом, связка MySQL и PHP предлагает гибкую систему для разработки и создания баз данных.

## Выборка

### Выборка по датам

Дата приёма		Дата примерки 1		Дата примерки 2		Дата сдачи	
С <input type="text" value="Выберите дат"/>	По <input type="text" value="Выберите дат"/>	С <input type="text" value="Выберите дат"/>	По <input type="text" value="Выберите дат"/>	С <input type="text" value="Выберите дат"/>	По <input type="text" value="Выберите дат"/>	С <input type="text" value="Выберите дат"/>	По <input type="text" value="Выберите дат"/>

### Выборка по другим вариантам

Клиника	Врач	Клиент	Техник	Вид работы
<input type="text" value="Выберите к"/>	<input type="text" value="Выберите в"/>	<input type="text" value="Выберите к"/>	<input type="text" value="Выберите т"/>	<input type="text" value="Выберите в"/>
<input type="button" value="Просмотр"/>				

### Результат

< 1 из 36 >

Действия	Номер	Клиника	Врач	Клиент	Техник	Работа	Цвет	Дата приёма	Дата примерки 1	Дата примерки 2	Дата сдачи	Срочность	Гарантия	Стоимость
	75	Лидер Дент	Зайцева Е.Ю.	Сергеенко	Елена Гулевич	Ортодонтическая пластинка		2015-01-12	0000-00-00	0000-00-00	2015-01-17	0		350000

### Результат

< 1 из 36 >

Действия	Номер	Клиника	Врач	Клиент	Техник	Работа	Цвет	Дата приёма	Дата примерки 1	Дата примерки 2	Дата сдачи	Срочность	Гарантия	Стоимость
	75	Лидер Дент	Зайцева Е.Ю.	Сергеенко	Елена Гулевич	Ортодонтическая пластинка		2015-01-12	0000-00-00	0000-00-00	2015-01-17	0		350000
	74	Авалон	Шачиков	Лапицкая	Елена Гулевич	Бигель ацетал	A3.5	2015-01-12	2015-01-15	0000-00-00	0000-00-00	0		1600000
	73	Улыбка Удачи	Зайцев	Сладченко В.Г.	Елена Гулевич	Бигель ацетал	A3.5	2015-01-12	2015-01-15	0000-00-00	0000-00-00	0		1600000
	72	Улыбка Удачи	Зайцев	Устинов Я.Е.	Юрий Гулевич	Пластмас	Д3	2015-01-12	0000-00-00	0000-00-00	2015-01-20	0		160000
	71	Клиника Боброва	Крюкова	Бугай А.М.	Тонкошурик	Цельнолит		2015-01-12	0000-00-00	0000-00-00	2015-01-16	0		210000
	70	Улыбка Удачи	Новиков	Исаенко С.Н.	Тонкошурик	Вкладка разборная		2015-01-12	0000-00-00	0000-00-00	2015-01-15	0		170000
	69	Жлобин	Стас	Лобин	Александр Погоряко	МК (Duceram Kiss, Vintage MP) никель	A3	2015-01-12	0000-00-00	0000-00-00	2015-01-14	0		330000
	68	Жлобин	Стас	Шевцов	Александр Погоряко	МК (Duceram Kiss, Vintage MP) никель	A3	2015-01-12	0000-00-00	0000-00-00	2015-01-14	0		1320000
	67	Жлобин	Стас	Восняков	Александр Погоряко	МК (Duceram Kiss, Vintage MP) никель	A2	2015-01-12	0000-00-00	0000-00-00	2015-01-14	0		1320000



Обнулить

### Добавить запись

Клиника	Врач	Клиент	Техник	Работа	Цвет	Дата приёма	Дата прим. 1	Дата прим. 2	Дата сдачи	Срочность	Гарантия
<input type="text" value="Выберите"/>	<input type="text" value="Выберите"/>	<input type="text"/>	<input type="text" value="Выберите"/>	<input type="text" value="Выберите"/>	<input type="text"/>	<input type="text" value="Выберите"/>	<input type="text" value="Выберите"/>	<input type="text" value="Выберите"/>	<input type="text" value="Выберите"/>	<input type="text" value="Выберите"/>	<input type="text"/>
<input type="button" value="Сохранить"/>											

Рисунок 2

**Н.В. Хазанова (УО «ГГУ им. Ф. Скорины», Гомель)**  
 Науч. рук. **Е.Е. Пугачёва**, ассистент

## РАЗРАБОТКА СТРУКТУРЫ БАЗЫ ДАННЫХ УЧЕТА УСПЕВАЕМОСТИ УЧАЩИХСЯ

Для разработки системы учета успеваемости учащихся необходимо хранить и обрабатывать информацию о преподавателе, о преподаваемых предметах, учитывать личные данные учащихся, группу и год обучения.

Для выставления оценки нужно учитывать преподавателя, который ее выставил. Значит, в базе данных нужно хранить информацию следующего характера:

- фамилия, имя, отчество преподавателя;
- преподаваемый предмет;
- категория преподавателя;
- стаж преподавателя.

Для учета оценок по конкретным предметам необходимо вести журнал предметов со следующими данными:

- название предмета;
- данные о преподавателе, который выставляет оценку по предмету.

Для выставления оценки учащемуся необходимо учитывать их личные данные:

- фамилия, имя, отчество учащегося;
- дата рождения учащегося;
- год поступления в учреждение образования;
- группа обучения (класс).

Для учета периода выставления оценки в базе данных нужно учесть следующее:

- группа обучения;
- год обучения группы.

Оценка, которая может быть выставлена учащемуся, представляет собой сложную структуру, поэтому ее следует отнести к отдельной категории:

- оценка по предмету либо сведения по аттестации.

Для выставления оценки общая информация содержит:

- группа обучения;
- год обучения;
- предмет;
- преподаватель по предмету;
- учащийся;
- оценка.

Для редактирования данных средствами SQL существуют 3 команды. Команда добавления записи в таблицу имеет следующий вид:

```
INSERT INTO <имя таблицы> [(<имя поля1>, <имя поля2>,...) ]  
VALUES(<значение1>, <значение2>, ...)
```

В разделе Values перечисляются значения, которые будут присвоены полям, а список полей после имени таблицы определяет, каким полям соответственно будут присвоены значения. Если список имен полей опущен (он не является обязательным), то значения будут

присваиваться полям в том порядке, в котором поля определены в структуре таблицы.

Команда удаления записей из таблицы:

```
DELETE FROM <имя_таблицы>  
[WHERE <условие_удаления_записей>]
```

Секция условия WHERE позволяет ограничить удаление только записями, удовлетворяющими заданному параметру. Если она опущена, из таблицы удалятся все записи.

Команда редактирования записей:

```
UPDATE <имя_таблицы>  
SET <название_поля1> = <выражение1>, <название_поля2> =  
<выражение2>,...  
[WHERE <условие>]
```

Команда позволяет изменить значения указанных полей для всех записей, либо для какого-то их подмножества (если используется условие в секции WHERE, которому должны удовлетворять изменяемые записи). В качестве выражений в разделе SET могут использоваться константы и выражения – в том числе и с участием значения самого поля по его имени.

Для разработки структуры базы данных в основу берется реляционная база данных со следующими принципами:

- данные воспринимаются пользователем как таблицы;
- каждая таблица состоит из однотипных строк и имеет уникальное имя;
- строки имеют фиксированное число полей (столбцов) и значений (множественные поля и повторяющиеся группы недопустимы). Иначе говоря, в каждой позиции таблицы на пересечении строки и столбца всегда имеется в точности одно значение или ничего;
- строки таблицы обязательно отличаются друг от друга хотя бы единственным значением, что позволяет однозначно идентифицировать любую строку такой таблицы;
- столбцам таблицы однозначно присваиваются имена, и в каждом из них размещаются однородные значения данных (даты, фамилии, целые числа или денежные суммы);
- полное информационное содержание базы данных представляется в виде явных значений данных, и такой метод представления является единственным;
- при выполнении операций с табколледж ее строки и столбцы можно обрабатывать в любом порядке безотносительно к их информационному

содержанию. Этому способствует наличие имен таблиц и их столбцов, а также возможность выделения любой их строки или любого набора строк с указанными признаками.

### **Литература**

1. Кузнецов, С.Д. Основы баз данных / С.Д. Кузнецов. – М.: Интернет-Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2007.
2. Анисимова, Ж.Н. Создание баз данных в СУБД Access / Ж.Н. Анисимова. – Мн.: БГУ, 1998.

**Н.В. Хазанова (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. Е.Е. Пугачёва, ассистент

### **ИСПОЛЬЗОВАНИЕ WEB-ПРИЛОЖЕНИЯ ДЛЯ АВТОМАТИЗАЦИИ УЧЕТА УСПЕВАЕМОСТИ УЧАЩИХСЯ КОЛЛЕДЖА**

SQL означает Structured Query Language – Структурированный Язык Запросов. Это – язык, который дает возможность создавать и работать в реляционных базах данных Стандарт SQL определяется с помощью кода ANSI (Американский Национальный Институт Стандартов).

В 1986 г. ANSI выпустил первый стандарт языка (ANSI X3.135-1986). В 1989 г. вышел следующий стандарт ANSI SQL-89. Этот же стандарт был принят ISO (ISO 9075-1989 «Database Language SQL with Integrity Enhancement»). В настоящее время стандарт ISO/IEC 9075 постоянно обновляется. Большая часть имеющихся средств использует стандарт SQL-92 и более поздние стандарты 90-х гг.

На основе стандарта в языке SQL выделяются следующие группы:

DDL (Язык Определения Данных) – так называемый Язык Описания Схемы в ANSI, состоит из команд, которые создают объекты (таблицы, индексы, просмотры, и так далее) в базе данных.

DML (Язык Манипулирования Данными) – это набор команд, которые определяют, какие значения представлены в таблицах в любой момент времени.

DCD (Язык Управления Данными) состоит из средств, которые определяют, разрешить ли пользователю выполнять определенные действия или нет.

Запрос выборки данных – команда, которая сообщает программе, управляющей базой данных, какую информацию требуется выбрать из таблиц в память. Эта информация обычно посылается непосредственно на экран компьютера или терминал, ее можно также послать принтеру, сохранить в файле или представить как вводную информацию для другой команды или процесса. Запросы выборки обычно рассматриваются как часть языка DML.

Эта команда запроса на выборку данных называется SELECT и имеет следующую структуру:

```
SELECT [ALL | DISTINCT] <имена полей>  
FROM <имена таблиц>  
[where <условие выборки>]  
[group by <Имена полей группировки>]  
[having <условие на группу>]  
[order by <имена полей сортировки> ASC | DESC]
```

Имена полей задаются через запятую. Если выборка производится из нескольких таблиц, то во избежание дублирования имен их указывают в виде <Имя таблицы>.<имя поля>. Если из таблицы выбираются все поля, то список имен полей можно заменить на «\*».

При наличии ключевого слова DISTINCT из таблицы, полученной применением списка выборки к результату табличного выражения, удаляются строки-дубликаты; при указании ALL (или просто при отсутствии DISTINCT) удаление строк-дубликатов не производится.

Раздел ORDER BY позволяет установить желаемый порядок: по возрастанию (ASC) или убыванию DESC – просмотра результата выражения запросов.

Раздел WHERE позволяет задать условие, по которому записи будут включаться или исключаться из результирующего запроса.

Раздел HAVING может осмысленно появиться в табличном выражении только в том случае, когда в нем присутствует раздел группировки GROUP BY. Условие поиска этого раздела задает условие на группу строк сгруппированной таблицы. Формально раздел HAVING может присутствовать и в табличном выражении, не содержащем GROUP BY. В этом случае полагается, что результат вычисления предыдущих разделов представляет собой сгруппированную таблицу, состоящую из одной группы без выделенных столбцов группирования. Результатом выполнения раздела HAVING является сгруппированная таблица, содержащая только те группы строк, для которых результат вычисления условия поиска есть true.

## Литература

1. Кузнецов, С.Д. Основы баз данных / С.Д. Кузнецов. – М.: Интернет-Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2007. – 484 с.
2. Анисимова, Ж.Н. Создание баз данных в СУБД Access [Текст] / Ж.Н. Анисимова. – Мн.: БГУ, 1998.

**А.Ю. Хачетлов (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **П.Л. Чечет**, канд. техн. наук, доцент

### **АВТОМАТИЗАЦИЯ ЗАПОЛНЕНИЯ ДОКУМЕНТА «ОТЧЁТ О РОЗНИЧНЫХ ПРОДАЖАХ» В 1С:ПРЕДПРИЯТИЕ**

На современном этапе развития общества автоматизация деятельности предприятия является одним из главных производственных активов предприятия. На предприятии любого масштаба необходимо вести точный учет кадров, оптимально распределять нагрузку между специалистами, мотивировать персонал на выполнение поставленной задачи, организовывать отлаженную работу, имеющую целью конечный результат, значимый для всего предприятия.

Эффективное руководство предприятием – это четко выстроенная система использования кадровых ресурсов предприятия. При наличии оптимального количества квалифицированных сотрудников такая система позволяет достигать поставленных целей в кратчайшие сроки и с минимальными затратами. Она помогает увеличивать продуктивность работы, расширять инновационную деятельность, повышать удовлетворенность клиентов, обеспечивать целостность бизнес-процессов.

Основная проблема, с которой сталкиваются сотрудники любого предприятия – это большая трудоемкость управления, огромное количество задач, функций, процессов, которыми необходимо оперативно и качественно управлять.

Все вышеперечисленные проблемы практически невозможно решить без применения современных средств сбора и обработки информации. Именно с использованием современных программных комплексов достигаются высокие результаты в скорости получения информации и удобства работы с ней. Весь этот процесс перехода от старых принципов работы к информационным технологиям называется автоматизацией.

Автоматизация заполнения документа предполагает создание общей системы, при которой осуществление операций было бы простым, быстрым и высокопроизводительным, а вывод документа на печать

позволяет сотруднику в кратчайшие сроки получить необходимые выходные данные.

Возможности автоматизации:

1. Получение руководством предприятия полной аналитической информации, необходимой для принятия решений.

2. Снижение затрат на основные HR-функции для руководителей службы персонала.

3. Автоматизация работы сотрудников.

4. Автоматическое формирование необходимых отчетов.

5. Автоматизация управления персоналом решает следующие задачи:

а) внедрение технологий для упрощения ведения оперативного учёта;

б) ведение и управление информацией о товарно-материальных ценностях.

Преимуществом автоматизации является снижение количества затраченного времени для получения необходимой информации, касаемой реестра розничных цен.

Таким образом, автоматизация заполнения документа и вывода его на печать является действенным инструментом, облегчающим не только действия бухгалтеру, но и обеспечивающим включенность каждого сотрудника в работу предприятия.

### **Литература**

1. Радченко, М.Г. 1С:Предприятие. Практическое пособие разработчика. Примеры и типовые приемы / М.Г. Радченко, Е.Ю. Хрусталева. – М.: ООО «1С-Публишинг», 2009. – 874 с.

2. Хрусталева, Е.Ю. Разработка сложных отчетов в 1С:Предприятие 8. Система компоновки данных / Е.Ю. Хрусталева. – М.: ООО «1С-Публишинг», 2009. – 513 с.

3. Коберн, Алистер. Современные методы описания функциональных требований к системам / Алистер Коберн – М.: Лори, 2002. – 288 с.

**А.Ю. Хачетлов (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **П.Л. Чечет**, канд. техн. наук, доцент

### **ВАЖНОСТЬ АВТОМАТИЗАЦИИ В ДЕЯТЕЛЬНОСТИ ПРЕДПРИЯТИЯ**

На современном этапе развития экономики автоматизация деятельности предприятия является одним из главных производственных



активов предприятия. На предприятии любого масштаба необходимо вести точный учет кадров, оптимально распределять нагрузку между специалистами, мотивировать персонал на выполнение поставленной задачи, организовывать отлаженную работу, имеющую целью конечный результат, значимый для всего предприятия.

Эффективное руководство предприятием – это четко выстроенная система использования кадровых ресурсов предприятия. При наличии оптимального количества квалифицированных сотрудников такая система позволяет достигать поставленных целей в кратчайшие сроки и с минимальными затратами. Она помогает увеличивать продуктивность работы, расширять инновационную деятельность, повышать удовлетворенность клиентов, обеспечивать целостность бизнес-процессов.

Автоматизация заполнения документа предполагает создание общей системы, при которой осуществление операций было бы простым, быстрым и высокопроизводительным, а вывод документа на печать позволяет сотруднику в кратчайшие сроки получить необходимые выходные данные.

Возможности автоматизации:

1. Получение руководством предприятия полной аналитической информации, необходимой для принятия решений.
2. Снижение затрат на основные HR-функции для руководителей службы персонала.
3. Автоматизация работы сотрудников.
4. Автоматическое формирование необходимых отчетов.

Преимуществом автоматизации является снижение количества затраченного времени для получения необходимой информации, касаемой реестра розничных цен.

Таким образом, автоматизация заполнения документа и вывода его на печать является действенным инструментом, облегчающим не только действия бухгалтеру, но и обеспечивающим включенность каждого сотрудника в работу предприятия.

**А.И. Хобня (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **О.М. Демиденко**, д-р техн. наук, профессор

## **ДЕКОМПОЗИЦИЯ СЕТЕЙ НОВОГО ПОКОЛЕНИЯ**

Моделирование сетей нового поколения NGN используется для оценки параметров работы сети в определенных условиях, оптимизации настроек оборудования и конфигурации сети. В частности,

при помощи моделирования можно определить следующие характеристики сети:

- предельная пропускная способность определенных фрагментов сети;
- характеристики работы сети при использовании различных топологий и протоколов маршрутизации;
- влияние числа пользователей того или иного сервера на показатели качества обслуживания (QoS);
- характеристики влияния мультимедийного трафика на работу сети;
- зависимость распределения количества потерь пакетов от нагрузки на отдельных узлах сети;
- и т. д.

Одним из важнейших этапов разработки имитационных моделей является процесс построения формальных моделей. Одним из методов формализации является *транзактный способ*, при котором объект моделирования представляется в качестве сети массового обслуживания (СМО).

Сеть массового обслуживания может включать в себя следующие статические элементы:

- генераторы заявок на обслуживание (*транзактов*);
- устройства обслуживания;
- элементы для накопления транзактов перед занятыми устройствами (как правило — очереди);
- потребители транзактов;
- дополнительные элементы.

Для формализации транзактным методом необходимо изучить следующие сведения об объекте моделирования:

- статическая структура моделируемой системы;
- динамическая структура моделируемой системы (дискретные события системы, потоки перемещения заявок между статическими узлами системы).

Основными *статическими элементами* сети NGN являются клиентские устройства, сервера, программные коммутаторы, шлюзы и каналы связи.

*Клиентскими устройствами* могут являться рабочие станции пользователей, IP-телефоны и иные устройства. С точки зрения сетей массового обслуживания данные устройства являются генераторами и потребителями транзактов. Для формализации статической структуры не существенными являются технические детали различий между клиентскими устройствами.

*Сервера* аналогично являются генераторами и потребителями транзактов с точки зрения теории СМО. Однако, они обрабатывают множество потоков транзактов от множества клиентов.

*Шлюзы* обеспечивают интеграцию сети NGN с сетями иной природы, например с ТФОП (телефонная сеть общего пользования). Таким образом, шлюзы также представляют собой генераторы и потребители заявок определенного типа.

*Каналы связи* осуществляют передачу данных между узлами сети. С точки зрения СМО, канал связи может быть представлен как *устройство обслуживания*. Основным параметром канала является его пропускная способность.

*Программные коммутаторы (Softswitch)* являются ключевыми устройствами сетей нового поколения и обеспечивают управление сеансами связи между другими узлами сети, используют технологии обеспечения качества обслуживания. Softswitch представляет собой устройство управления сетью NGN, которое позволяет отделить функции управления сеансами связи от функций коммутации пакетов, как правило, обслуживающее большое число абонентов и взаимодействующее с серверами приложений. Программные коммутаторы также могут быть представлены в качестве устройств обслуживания. *Механизмы обеспечения качества обслуживания (QoS)* с точки зрения СМО являются элементами накопления транзактов перед устройствами обслуживания. При этом механизмы обеспечения QoS имеют сложное устройство и в большинстве случаев не являются реализацией обычных очередей. Кроме того часть пакетов может уничтожаться в рамках работы механизма QoS. Поэтому данные механизмы также являются и потребителями заявок.

Рассмотрим *динамическую структуру* объекта моделирования. Основным динамическим компонентом сетей NGN являются пакеты. С точки зрения теории СМО пакеты могут быть представлены в качестве транзактов. Для целей разработки имитационных моделей необходимо исследовать виды, размеры, важнейшие особенности генерации, обработки, скорость передачи и некоторые другие свойства пакетов.

Важнейшим типом пакетов в сети NGN являются пакеты голосового трафика (VoIP). Для моделирования генерации и передачи VoIP-пакетов на низком уровне необходимо определить следующие характеристики:

- размер голосовых пакетов;
- частота генерации пакетов;

Для определения первой характеристики необходимо исследовать особенности конкретного протокола передачи данных и используемого

алгоритма сжатия. Для определения второй характеристики необходимо исследовать сведения о распределении продолжительности сеансов связи, интервалов между ними, интервалов генерации пакетов внутри сеанса и другие особенности. Для целей моделирования необходимо определить особенности генерации пакетов и других популярных типов трафика, например HTTP, FTP, BitTorrent и т. д.

Таким образом, используя транзактный способ, осуществляется построение формальной модели сети нового поколения NGN. Затем формальная модель используется для разработки имитационной модели. Детали озвучены в докладе.

**А.И. Хобня (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **О.М. Демиденко**, д-р техн. наук, профессор

## **РАЗРАБОТКА УНИВЕРСАЛЬНОЙ ПЛАТФОРМЫ МОДЕЛИРОВАНИЯ МЕХАНИЗМОВ ОБЕСПЕЧЕНИЯ КАЧЕСТВА ОБСЛУЖИВАНИЯ В СЕТЯХ ПАКЕТНОЙ ПЕРЕДАЧИ ДАННЫХ**

Моделирования работы механизмов обеспечения качества обслуживания в сетях пакетной передачи данных может осуществляться в следующих целях:

- оценка значений показателей качества обслуживания при использовании различных настроек для исследования резерва ресурсов сети;
- оптимизация настроек механизма обеспечения качества обслуживания для улучшения характеристик работы сети;
- исследование степени влияния трафика различных классов на работу сети;
- определение пороговой нагрузки на узлах сети;
- исследование различных свойств существующих и разработка новых механизмов обеспечения качества обслуживания.

Современные инструментальные средства моделирования сетей с коммутацией пакетов, в частности NS-3, поддерживают моделирование некоторых механизмов обеспечения качества обслуживания (QoS). Для моделирования некоторых специфических механизмов и сценариев используются специальные платформы, например AQM&DoS Simulation Platform. Для построения универсальной платформы моделирования механизмов обеспечения QoS, которая позволит исследовать свойства как существующих механизмов, так и разрабатываемых, необходим обобщенный метод представления механизмов обеспечения QoS.

Анализ алгоритмов работы механизмов обеспечения качества обслуживания, позволяет выделить четыре основных компонента каждого механизма обеспечения QoS:

- 1) алгоритм классификации пакетов;
- 2) очереди пакетов;
- 3) алгоритмы активного управления очередью (AQM);
- 4) алгоритм планирования (network scheduling).

*Алгоритмы классификации пакетов* осуществляют определение класса пакета. Пакеты различных классов распределяются по различным очередям. Проанализировав алгоритмы классификации необходимо отметить следующие их основные свойства: использование специальных полей IP-пакетов, предназначенных для маркировки типа трафика (ToS, DSCP, MPLS QoS), использование IP-адреса отправителя, IP-адреса получателя, порта отправителя и порта получателя, а также использование содержимого пакета для получения информации о прикладных протоколах и приложениях.

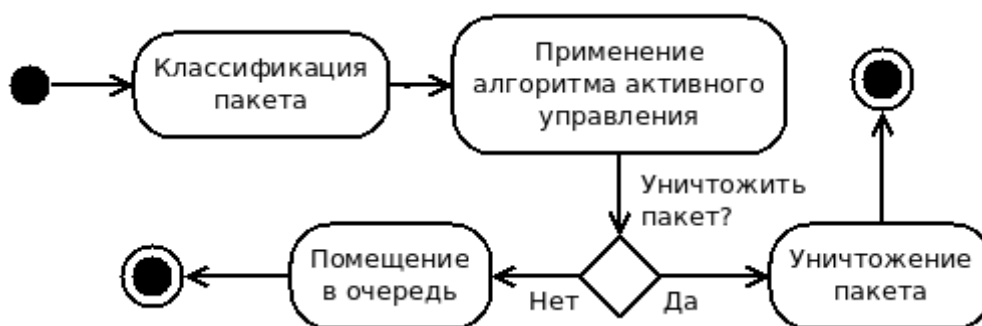


Рисунок 1 – диаграмма активностей поведения механизма QoS при прибытии входящего пакета

*Алгоритмы активного управления очередью* (Active queue management – AQM) осуществляют управление уничтожением (или маркированием) пакетов сетевого интерфейса, в случаях переполнения или достижения состояния близкого к переполнению внутреннего буфера устройства. Эти действия производятся в целях сокращения нагрузки и обеспечения качества обслуживания. Для алгоритмов активного управления очередями характерны следующие особенности: наличие настраиваемых параметров, управление единственной очередью, использование данных о размере буфера очереди и его занятом объеме, использование данных об отправителе, получателе, иных полях заголовка, сохранение внутреннего состояния. Следует заметить, что для каждой очереди в механизме обслуживания может использоваться отдельный алгоритм активного управления.



Рисунок 2 – Диаграмма активностей поведения механизма QoS при готовности отправки следующего пакета

*Алгоритмы планирования* осуществляют управление последовательностью отправки пакетов. Алгоритмы планирования обладают следующими особенностями: наличие настраиваемых параметров, управление всеми очередями механизма обеспечения QoS, хранение и изменение внутреннего состояния.

Для создания имитационной модели механизма обеспечения качества обслуживания необходимо смоделировать его работу при наступлении событий двух типов: прибытие входящего пакета, готовность исходящего интерфейса принять пакет. Диаграммы активностей механизма обеспечения QoS при наступлении данных событий изображены на рисунках 1 и 2. Для описания механизма обеспечения QoS необходимо задать каждый из его компонентов. Алгоритмы классификации, планирования и активного управления очередями представляются в виде функций от определенных параметров, таких как свойства прибывшего или отправляемого пакета, состояние очередей и внутренне состояние. С точки зрения объектно-ориентированного программирования, каждый из компонентов представляет собой объект, реализующий строго определенный интерфейс. Технические детали реализации данного подхода представлены в докладе.

**Ю.В. Хомицкий (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **П.Л. Чечет**, канд. техн. наук, доцент

## **АВТОМАТИЗАЦИЯ СКЛАДСКОГО УЧЕТА**

Автоматизация организаций оптовой торговли базируется на выделении функциональных подсистем и задач, которые соответствуют ее хозяйственной деятельности. В соответствии с разделением функций

между подразделениями предприятия в системах автоматизации организаций оптовой торговли выделяют две функциональные подсистемы:

- складской учет товаров;
- бухгалтерский учет товаров.

Для учета товаров применяются различные бухгалтерские счета, связанные с переходом товара из одного состояния в другое.

В информационных системах с точки зрения автоматизации существуют только два объекта – сущности и процессы.

Сущности взаимосвязаны между собой и являются узловыми точками сбора информации. Они соответствуют понятию «объекты», из которых состоит информационная система.

Процессы – преобразование одних сущностей в другие, либо движение сущностей во времени и пространстве, в которых могут участвовать другие сущности. Например, в процессе поступления товара на склад организации принимают участие, как минимум, сущности товар и поставщик.

Счета бухгалтерского учета являются способом вторичной регистрации товаров и операций с ними. На них товары фиксируются в денежном измерении. Процессы, происходящие с товарами и другими средствами, приводят либо к их уменьшению, либо к увеличению. С целью отдельного учета указанных процессов по каждому счету учет ведется по дебету и кредиту. По дебету отражается увеличение средств, по кредиту – уменьшение.

Для учета движения товаров составляют бухгалтерские проводки, которые отражают вид движения товаров: поступление товаров на склад, отгрузка, реализация, инвентаризация и другие.

Проводки (хозяйственные операции) отражаются на счетах с помощью двойной записи, которая показывает взаимную связь объектов учета. Двойная запись, по сути, показывает внутреннюю связь объектов, экономический смысл и содержание каждой операции, переход средств из одного состояния в другое. В бухгалтерском учете взаимосвязь счетов в проводке называют корреспонденцией, а счета, составляющие проводку, корреспондирующийся.

**Ю.В. Хомицкий (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **П.Л. Чечет**, канд. техн. наук, доцент

## **АВТОМАТИЗАЦИЯ УЧЕТА МАТЕРИАЛОВ НА СКЛАДЕ**

Современные системы автоматизации учета товаров должны удовлетворять различным требованиям, среди которых одним из важнейших

является требование функциональной полноты. Функциональная полнота системы автоматизации подразумевает наличие в программной системе необходимого и достаточного количества компонентов, которые:

1 наиболее полно соответствуют информационным потребностям пользователей;

2 автоматизируют все функции, охватывают все виды деятельности конкретной предметной области.

Автоматизированная система должна соответствовать требованиям предметной области, поскольку они влияют как на выделяемые в средствах автоматизации функциональные подсистемы, так и на состав задач, их взаимосвязь, алгоритмы обработки информации. Оценка систем автоматизации любых объектов экономического управления, в том числе организаций оптовой торговли, с позиций функциональной полноты базируется на знании особенностей предметной области, особенностей обработки информации.

Автоматизация организаций оптовой торговли базируется на выделении функциональных подсистем и задач, которые соответствуют ее хозяйственной деятельности. В соответствии с разделением функций между подразделениями предприятия в системах автоматизации организаций оптовой торговли выделяют две функциональные подсистемы:

- складской учет товаров;
- бухгалтерский учет товаров.

Указанные подсистемы могут входить в комплексные системы автоматизации торгового предприятия, как отдельные программные модули, взаимосвязанные между собой в процессе обработки информации.

В оптовой торговле решают задачи:

- учет поступления товаров в местах хранения;
- отчетность материально ответственного лица;
- контроль за количеством и качеством товара в местах хранения;
- контроль правильности составления документов в местах хранения товаров;
- учет поступления товаров в бухгалтерском учете;
- формирование цены товара;
- учет отгрузки, реализации товаров;
- учет товарных запасов;
- учет налога на добавленную стоимость;
- учет тары;
- инвентаризация товаров;
- отражение результатов инвентаризации в бухгалтерском учете;
- учет расчетов с поставщиками и подрядчиками;
- инвентаризации товаров.



Для учета товаров применяются различные бухгалтерские счета, связанные с переходом товара из одного состояния в другое.

В информационных системах с точки зрения автоматизации существуют только два объекта – сущности и процессы.

Сущности взаимосвязаны между собой и являются узловыми точками сбора информации. Они соответствуют понятию «объекты», из которых состоит информационная система. Так, в системе автоматизации организаций оптовой торговли сущностями, как узловые точки сбора информации, являются:

- товар, который определяется наименованием, единицей измерения, ценой, сортом, артикулом и т. д.;

- поставщик (покупатель) определяется наименованием, адресом, расчетным счетом и другими реквизитами и т. д.

Процессы – преобразование одних сущностей в другие, либо движение сущностей во времени и пространстве, в которых могут участвовать другие сущности. Например, в процессе поступления товара на склад организации принимают участие, как минимум, сущности товар и поставщик.

Это позволяет оценить различие товара, хранящегося на складе, и товара, учет которого ведется в бухгалтерии. Товар на складе определяется следующими характеристиками:

- наименование;

- сорт;

- цена;

- единица измерения, количество (объем);

- стоимость;

- таможенная декларация и другие реквизиты.

Указанные характеристики являются исходной информацией для автоматизированного учета товара.

Счета бухгалтерского учета являются способом вторичной регистрации товаров и операций с ними. На них товары фиксируются в денежном измерении. Процессы, происходящие с товарами и другими средствами, приводят либо к их уменьшению, либо к увеличению. С целью отдельного учета указанных процессов по каждому счету учет ведется по дебету и кредиту. По дебету отражается увеличение средств, по кредиту – уменьшение.

Для учета движения товаров составляют бухгалтерские проводки, которые отражают вид движения товаров: поступление товаров на склад, отгрузка, реализация, инвентаризация и другие.

Проводки (хозяйственные операции) отражаются на счетах с помощью двойной записи, которая показывает взаимную связь объектов

учета. Двойная запись, по сути, показывает внутреннюю связь объектов, экономический смысл и содержание каждой операции, переход средств из одного состояния в другое. В бухгалтерском учете взаимосвязь счетов в проводке называют корреспонденцией, а счета, составляющие проводку, корреспондирующийся.

Таким образом, бухгалтерская проводка представляет собой корреспонденцию счетов, когда одновременно делается запись по дебету и кредиту счетов на сумму хозяйственной операции, подлежащей регистрации.

### Литература

1. Радченко, М.Г. 1С:Предприятие 8.2. Практическое пособие разработчика / М.Г. Радченко, Е.Ю. Хрусталёва. – М.: ООО «1С – Паблишинг», 2009. – 872 с.

2. Тимофеев, Г.С. Конфигурирование и администрирование 1С:Предприятие / Г.С. Тимофеев, Д.С. Шумейко. – М.: Феникс, 2003. – 320 с.

3. Когаловский, М.Р. Энциклопедия технологий баз данных / М.Р. Когаловский. – М.: Финансы и статистика, 2003. – 800 с.

4. Рязанцева, Н.А. 1С:Предприятие. Бухгалтерский учет. Секреты работы / Н.А. Рязанцева, Д.Н. Рязанцев. – СПб.: БХВ – Петербург, 2004. – 350 с.

**Д.Е. Храбров (УО «ГГТУ им. П.О. Сухого», Гомель)**

Науч. рук. **И.А. Мурашко**, д-р техн. наук, доцент

### МЕТОДИКА ЛОКАЛЬНОГО WI-FI ПОЗИЦИОНИРОВАНИЯ НА ОСНОВАНИИ АЛГОРИТМА СОПОСТАВЛЕНИЯ С ОБРАЗЦОМ

GPS навигация не подходит для многоэтажных зданий, а значит необходимо использовать другую технологию. Предлагается методика позиционирования объекта внутри организаций на основе *Wi-Fi*.

Основные положения предлагаемой методики локального позиционирования можно разбить на два этапа [1]. Предварительный этап – этап обучения системы, ведь для расчёта необходимо иметь эталонный набор точек с уже известными координатами. На этом этапе необходимо получить опорные координаты для каждого помещения:

$$Room_i = ( [P_1, Name_1], [P_2, Name_2] \dots, [P_m, Name_m] ).$$

То есть для каждой комнаты необходимо хранить все сигналы видимых точек доступа ( $P_j$ ) и имена этих точек ( $Name_j$ ).

Далее необходимо сравнить список видимых точек со списком разрешённых, чтобы отфильтровать неразрешённые:

$$Rm_i = allowed\{ Room_i \}.$$

Далее остаётся лишь сохранить в базу данных информацию о помещении, содержащую информацию о пяти наиболее сильных источниках для данного помещения:

$$DB_{Room_i} = mazZ_1^M \{ Rm_i \}.$$

По умолчанию выбирается  $Z = 5$ , то есть берутся 5 наиболее мощных точек доступа. Однако этот параметр трудно задать фиксировано, так как он зависит от многих факторов. Например, в спортзале желательно сохранять 7 точек доступа, а в учебных помещениях с большим количеством *Wi-Fi* точек доступа будет достаточно хранить 3. Основной этап работы системы представлен на блок-схеме, рисунок 1. Пронумерованные блоки рисунка 1:

1) Устройство снимает уровни сигнала до опорных точек и идентифицирует эти точки ( $Room_i$ ).

2) Отбрасываются точки с низким уровнем и проверяется, как много точек доступа имеют достаточный уровень сигнала ( $Rm_i$ ). Это действие выполняется 3–5 раз за короткий промежуток времени. Сильно отличающиеся от других результаты эксперимента отбрасываются, среди оставшихся одноимённые координаты усредняются.

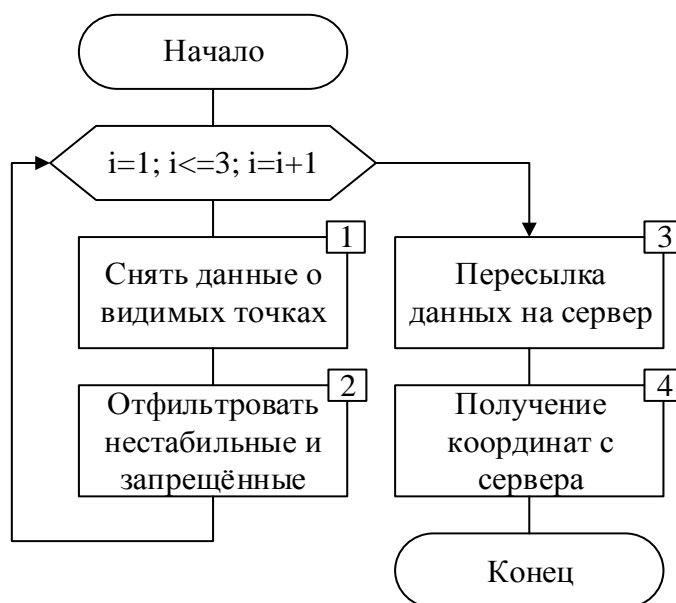


Рисунок 1 – Блок-схема работы клиента

3) Пересылка координат на сервер, согласно графику расписания звонков. Кроме статически указанных интервалов поддерживаются и динамические временные интервалы (гибкий график).

4) Сервер обрабатывает полученные данные и определяет, в каком конкретно помещении находится данный объект. Статистика хранится на сервере для последующего использования.

Слабым местом методики является необходимость переобучения системы в случае реконфигурации оборудования. Например, изменение типа или местоположения точек доступа, добавление новых точек с высоким уровнем сигнала и тому подобное. Таким образом, после каждой реконфигурации требуется корректировка информации для позиционирования. В некоторых случаях это может быть эквивалентно первоначальному обучению системы [2]. Для решения данной проблемы предлагаем использовать так называемый динамический эталон.

Кроме непосредственно самой методики следует учесть несколько дополнений к ней.

1. Необходимо делать три контрольных измерения. Если значения хотя бы двух измерений совпали, то положение найдено. Это отображено в блоке 1 на рисунке 1.

2. При неудачном измерении координат в текущий момент времени по трём измерениям необходимо информировать сервер статистики о том, что положение в данный момент времени не известно.

3. Имеет смысл помнить о неточностях алгоритма – если стоять в аудитории возле стены, то в качестве координат можно получить координаты соседнего помещения.

4. Период снятия данных должен быть выставлен в соответствии с расписанием занятий: во время пар снимать данные раз в полчаса, во время перемен – раз в пять минут.

После введения в методику приведенных уточнений были получены результаты, показанные на рисунке 2. Каждые 1000 секунд производилось 3 измерения текущей координаты, которые затем округлялись по точности до помещения. Если хотя бы 2 измерения из 3 совпадают, то помещение считается корректно найденным.

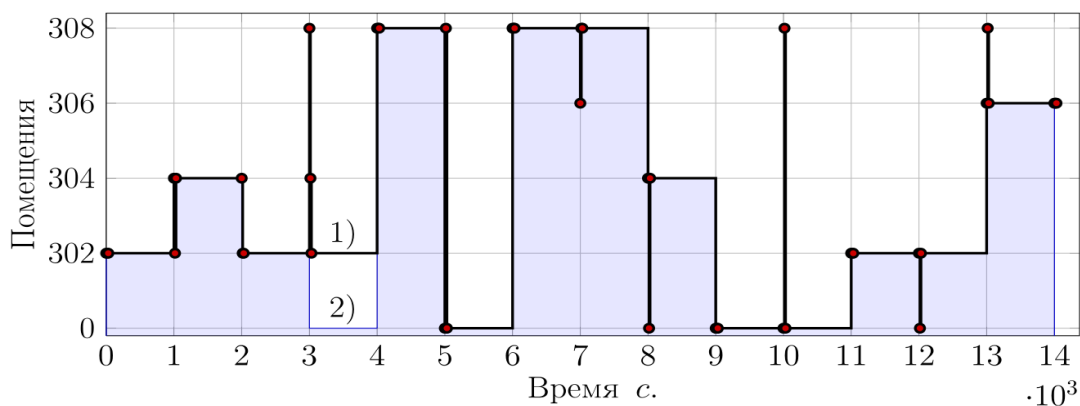


Рисунок 2 – График положения от времени:  
1) вычисленное положение; 2) реальное положение

В данной работе предложена методика позиционирования внутри помещения на основании существующей *Wi-Fi* сети. Применение данной методики позволяет оперативно получить информацию о положении студента или преподавателя в ВУЗе (используя программное обеспечение сервера, который собирает статистику), а также вести автоматизированный учёт посещения занятий для модульно-рейтинговой системы.

### Литература

1. Храбров, Д.Е. Методика позиционирования и контроля посещаемости студентов на основании WiFi-сети университета / Д.Е. Храбров, И.А. Мурашко / М-во образования Респ. Беларусь, Бел. гос. ун-т. инф-ки и рад-ки. – Минск, 2014. – С. 254–255.

2. A survey of mathematical methods for indoor localization / F. Seco, A.R. Jimenez, C. Prieto et al. // Intelligent Signal Processing, 2009. WISP 2009. IEEE International Symposium on. – 2009. – P. 9–14.

**А.Н. Цябус (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч.рук. **Г.Л. Карасёва**, канд. физ.-мат. наук, доцент

### ИСПОЛЬЗОВАНИЕ UNITY3D ДЛЯ СОЗДАНИЯ ИГРОВОГО ПРИЛОЖЕНИЯ

Unity3d является современным кроссплатформенным движком для создания игр и приложений. Разработкой занимались Unity Technologies. С помощью данного движка можно разрабатывать не только приложения для компьютеров, но и для мобильных устройств (например, на базе Android, IOS), игровых приставок и других девайсов.

К основным характеристикам движка стоит отнести то, что в среде разработки Unity интегрирован игровой движок, иными словами, присутствует возможность протестировать игровое приложение не выходя из редактора. Во-вторых, Unity поддерживает импорт огромного количества различных форматов, что позволяет разработчику игры конструировать сами модели в более удобном приложении, а Unity использовать по прямому назначению – разрабатывать продукт. В-третьих, написание сценариев (скриптов) осуществляется на наиболее популярных языках программирования – C# и JavaScript.

Unity – это мультиплатформенный инструмент для разработки двух- и трёхмерных приложений и игр, работающий под операционными системами Windows и OS X. Созданные с помощью Unity приложения

работают под операционными системами Windows, OS X, Windows Phone, Android, Apple iOS, Linux, а также на игровых приставках Wii, PlayStation 3 и Xbox 360. Приложения, созданные с помощью Unity, поддерживают DirectX и OpenGL.

Свойства объектов настраиваются несколькими кликами мыши, чтобы назначить текстуры, звук, поведение, скрипты и т. д. Есть возможность клонирования объектов (иногда нужно перенести или размножить сложные игровые объекты). Игровые объекты полностью управляемы и настраиваются.

Таким образом, Unity3d является актуальной платформой, с помощью которой можно создавать свои собственные приложения и экспортировать их на различные устройства, будь то мобильный телефон или приставка Nintendo Wii.

Для того чтобы создать игру, как минимум, нужно владеть одним из доступных (на Unity) языков программирования: C#, JavaScript или Boo.

Сервер ресурсов доступен как для Mac OS X Installer, так и для Linux RPMs. Поддержка нескольких платформ дает гибкость в том, как внедрить Сервер ресурсов Unity в существующую IT-инфраструктуру.

**А.Н. Цябус (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч.рук. **Г.Л. Карасёва**, канд. физ.-мат. наук, доцент

## **РАЗРАБОТКА САЙТА НА CMS JOOMLA С ИСПОЛЬЗОВАНИЕМ РАСШИРЕНИЙ**

На сегодняшний день основным источником информации, несомненно, является Интернет – всемирная система объединённых компьютерных сетей, построенная на базе IP и маршрутизации IP-пакетов.

Изготовление сайтов как работающих целостных информационных ресурсов и систем – это составной процесс, вовлекающий труд различных специалистов. Данный вид деятельности носит название веб-разработка. Владельцы будущего сайта разрабатывают сайты своими силами, либо обращаются к специализированным разработчикам. Создание сайта состоит из нескольких этапов, таких как: разработка дизайна, вёрстка, программирование и обеспечение безопасности.

Для ускорения и облегчения процесса разработки можно воспользоваться CMS. Система управления содержимым (англ. Content management system, CMS) – информационная система или компьютерная программа, используемая для обеспечения и организации совместного

процесса создания, редактирования и управления контентом. Joomla является свободно доступным программным обеспечением с открытым кодом (FOSS), и поэтому его исходный код включается в обычный загружаемый пакет. В качестве дополнительного инструментария используют фреймворки.

Фреймворк – структура программной системы; программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта. Это своеобразный скелет, а также набор различных инструментов, с помощью которых можно создать неповторимый шаблон.

Расширения Joomla – это объединённое наименование всех компонентов, модулей, плагинов, шаблонов и т.д. Расширением можно назвать любую программу, которая устанавливается на сайт.

Программирование на Joomla заключается главным образом в написании кода на языке PHP. Информация для веб-сайта на Joomla хранится в базе данных, и поэтому часть этого кода PHP взаимодействует с базой данных. Как правило, такое взаимодействие осуществляется в форме запросов SQL. Запросы базы данных служат для сохранения и извлечения из неё данных.

Частично программирование для Joomla заключается и в работе с XML или HTML –документами.

**М.В. Черёмухин (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **А.Н. Крайников**, ассистент

## **2С:УПРАВЛЕНИЕ ТОРГОВЛЕЙ**

Программа «2С:Управление торговлей» (в дальнейшем «Мвух») представляет собой базу данных, призванную упростить работу по учёту товара при его реализации и помочь с расчётом налогов, при автоматическом заполнении книги учёта доходов и расходов (КУДиР). Мвух больше подходит при работе по упрощённой системе налогообложения в Республики Беларусь.

Программа позволяет:

- вести учёт товара;
- рассчитывать остатки, книгу учёта доходов и расходов, печатать ценники;
- вести учёт оказываемых услуг;
- вести книгу кассира;
- отписывать товар по накладной;

– соединяться с клиент-банком BSDClient 3, брать данные о поступлениях.

Ввод данных в электронную накладную можно осуществить несколькими способами. Запустите программу, и перейдите на вкладку «ГТН (ТН)», это и есть электронное отображение будущей накладной. Пока в ней нет записей, поэтому сперва создадим саму накладную, а затем заполним её. Создать накладную можно в окне выбора накладной. Чтобы его вызвать нажмите меню «Файл→Открыть». Окно выбора накладной имеет некоторые инструменты для работы с накладными. Эти инструменты выполнены в виде кнопок «Добавить, Применить, Удалить». Накладные отображаются по убыванию (сортировка по дате), т. е. самые новые вверху. Чтобы добавить накладную введите её номер в поле, расположенное под столбцом «Наименование» и укажите дату. Эти данные находятся непосредственно на самой накладной (Номер и дата) вверху. Введём номер накладной «111111» и укажем дату «01.04.2012». Нажмём кнопку «Добавить» и накладная появится. Дважды щёлкнем по ней мышкой, чтобы открыть созданную накладную. Признаком открытия накладной является отображения соответствующего номера и даты накладной в БД.

Перейдём непосредственно к заполнению накладной. Все данные вводятся с бумажного варианта накладной. Необходимо следить за соответствием столбцов, т. е. наименование товара вводить в столбец «Наименование товара», а НДС – в столбец «НДС%», причём в процентах. Так же следует отметить, что все данные вводятся непосредственно с клавиатуры, либо путём вставки из MS Excel с помощью контекстного меню «Вставить».

При вставке таблицы из MS Excel строки добавляются автоматически. Также необходимо следить за правильностью расположения столбцов в MS Excel и базе, чтобы столбцы не накладывались друг на друга, либо не вставлялись в несоответствующие поля. При заполнении накладной вручную, добавление строк осуществляется кнопкой «Добавить стр.», причём можно указать количество добавляемых строк (слева от кнопки «Добавить стр.»). Вставку из MS Excel удобно использовать, если Вы получаете в электронном варианте накладную или счёт. Тогда заполнение накладной занимает не более 10 минут. Также не стоит забывать про кнопки , которые помогают быстро заполнить столбцы «Сумма надб.» и «НДС%». Эти данные берутся из настроек программы (вкладка «Опции»), к ним мы вернёмся позднее. Кнопка «Подогнать» помогает расценить товар.

После сохранения, каждому товару присваивается уникальный код (Столбец “id”). В дальнейшем по этому коду можно осуществлять



поиск товара в базе. Так же этот код печатается на чеке, и его можно писать на товаре, чтобы знать из какой накладной этот товар к Вам приехал. Все настройки производятся на вкладке «Опции». Тут задаются НДС и розничная надбавка по умолчанию, для заполнения накладной.

Опции соединения программы с БД MySQL (Заполняются только опытными пользователями). Тут же указывается пароль администратора базы. Некоторые функции базы закрыты для повышения безопасности и защиты от порчи данных. Чтобы эти функции открыть, необходимо перейти в режим администратора и указать задаваемый тут пароль «Пароль Админа».

Вход в режим администратора осуществляется через меню «Инструменты». Будьте осторожны! Не рекомендуется работать в этом режиме, если Вы конкретно не представляете что делаете. Связь с клиент-банком осуществляется путём указания пути к выгрузкам этого клиента. Это осуществляется из вкладки РРЦ путём нажатия на кнопку «На печать». Шаблон РРЦ можно менять, для этого откройте файл шаблона, расположенный в папке «shab» в корневом каталоге программы. Путь к шаблону: Mbux\shab\RRC.dot

После изменения шаблона его необходимо сохранить как шаблон под тем же именем. При изменении шаблона рекомендуется не изменять специальные теги шаблона заключённые в {фигурные скобки}. Самая часто используемая часть базы это склад! Тут Вы наблюдаете за количеством товара, продаете его, списываете, ищите и печатаете ценники. Разберём элементы склада по порядку, чтобы было понятнее как с ним работать.

Быстрый поиск – группа параметров, задаваемых для выборки (поиска) данных на складе. Тут можно искать товар по его коду, части наименования, накладной, статусу, серийному номеру либо по заметкам (Доп. инф.).

Продажа – группа элементов, которые помогут Вам продать, либо списать товар. Тут указывается тип расчёта: наличный; безналичный; кредит; на собственное потребление (т. е. списать), количество единиц, для продажи, а так же дата продажи.

Чтобы продать товар, необходимо выбрать его на складе, заполнить параметры продажи и нажать кнопку «Продать» (либо F6 на клавиатуре).

Группа элементов для редактирования товара. Т. е. изменения его наименования, статуса, записи пометок. Некоторые опции доступны только в режиме администрирования. Изменённое наименование товара сохраняется только на складе, в накладной имя остается прежним.

Кнопка «В корзину» забрасывает одну единицу товара в корзину (о ней чуть позже), а кнопка «Серийники» позволяет заполнить серийные номера для каждой единицы товара (если их несколько). Допустим, Вы купили два телевизора, которые в накладной идут одной строкой с заданным количеством два. В этом случае в накладной Вы серийный номер не указываете, т. к. их два (у каждого телика свой уникальный серийный номер), а задаёте их тут, на складе. Просто выбрав товар и нажав кнопку «Серийники». Столбец «Серийник» заполняем, «Комментарий» – по желанию, а «Статус» не трогаем (заполнится автоматически). Тут строки добавляются автоматически.

При поиске товара на складе по серийнику, поиск производится по всем полям. Т. е. и тем, что в накладной, и тем, что на складе. Многие операции дублируются контекстным меню (По щелчку правой кнопкой мышки на товаре в списке).

Программа поддерживает неограниченное количество складов, перемещать товар по складам можно с помощью контекстного меню в складе, выбирается склад в группе элементов «Быстрый поиск».

**А.И. Чернышев (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **А.И. Кучеров**, ст. преподаватель, магистрант техн. наук

## **ОСНОВНЫЕ ВОЗМОЖНОСТИ СИСТЕМЫ «1С:ПРЕДПРИЯТИЕ 8.2» И ОБЛАСТИ ЕЁ ПРИМЕНЕНИЯ**

С развитием информационных технологий компьютеры, с их расширенными функциональными возможностями, активно применяются в различных сферах человеческой деятельности, связанных с обработкой информации, представлением данных. В современном обществе, которое функционирует в жестких рыночных условиях, своевременная обработка информации способствует совершенствованию организации производства, оперативному и долгосрочному планированию, прогнозированию и анализу хозяйственной деятельности, что позволяет успешно конкурировать на рынке. Эти задачи можно решить с использованием автоматизированных информационных систем.

Использование баз данных и информационных систем становится неотъемлемой составляющей деловой деятельности современного человека и функционирования организаций. Поэтому среда разработки «1С:Предприятие 8.2» является универсальной системой автоматизации деятельности предприятия. За счет своей универсальности система «1С:Предприятие 8.2» может быть использована для автоматизации

самых разных участков экономической деятельности организации: учета товарных и материальных средств, взаиморасчетов с контрагентами и т. д.

Функционирование системы делится на два процесса:

– разработка (описание модели предметной области средствами системы);

– исполнение (обработка данных предметной области).

На этапе разработки система оперирует такими универсальными понятиями (объектами), как документ, журнал документов, справочник, реквизит, форма, регистр и другие. Совокупность этих понятий и определяет концепцию системы.

В процессе исполнения система уже оперирует конкретными понятиями, описанными на этапе конфигурирования (справочниками товаров и контрагентов, счетами, накладными и т. д.). При работе пользователя в режиме «1С:Предприятие 8.2» обработка информации выполняется как штатными средствами системы, так и с использованием алгоритмов, созданных на этапе конфигурирования.

Программа «1С:Сервисный центр» разработана для комплексной автоматизации сервисного центра специализирующегося на ремонте и обслуживании офисной техники. Данное отраслевое решение позволяет обрабатывать большие объемы информации, предоставить удобный интерфейс для работы с данными, позволяет пользователю, как просматривать необходимую информацию, так и, по мере необходимости, манипулировать ею.

**А.И. Чернышев (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **А.И. Кучеров**, ст. преподаватель, магистрант техн. наук

## **АВТОМАТИЗАЦИЯ СЕРВИСНОГО ЦЕНТРА В СРЕДЕ РАЗРАБОТКИ «1С:ПРЕДПРИЯТИЕ 8.2»**

С развитием информационных технологий компьютеры, с их расширенными функциональными возможностями, активно применяются в различных сферах человеческой деятельности, связанных с обработкой информации, представлением данных. В современном обществе, которое функционирует в жестких рыночных условиях, своевременная обработка информации способствует совершенствованию организации производства, оперативному и долгосрочному планированию, прогнозированию и анализу хозяйственной деятельности, что позволяет успешно конкурировать на рынке. Эти задачи можно решить с использованием автоматизированных информационных систем.

Использование баз данных и информационных систем становится неотъемлемой составляющей деловой деятельности современного человека и функционирования организаций. Поэтому среда разработки «1С:Предприятие 8.2» является универсальной системой автоматизации деятельности предприятия.

1С:Предприятие – это специализированная объектно-ориентированная система управления базами данных (СУБД), предназначенная для автоматизации деятельности предприятия.

Платформа «1С:Предприятие 8.2» представляет собой программную оболочку над базой данных. Кроме того, с версии 8.1 хранение данных возможно в СУБД PostgreSQL и IBM DB2, а с версии 8.2 добавилась и Oracle. Имеет свой внутренний язык программирования, обеспечивающий, помимо доступа к данным, возможность взаимодействия с другими программами посредством OLE и DDE технологий. Клиентская часть платформы функционирует только в среде ОС Microsoft Windows. Начиная с версии 8.1, серверная часть платформы в клиент-серверном варианте работы «1С: Предприятия» может функционировать на ОС Linux.

Гибкость платформы позволяет применять 1С:Предприятие в самых разнообразных областях:

- автоматизация производственных и торговых предприятий, бюджетных и финансовых организаций, предприятий сферы обслуживания и т. д.;

- поддержка оперативного управления предприятием;

- автоматизация организационной и хозяйственной деятельности;

- ведение бухгалтерского учета с несколькими планами счетов и произвольными измерениями учета, регламентированная отчетность;

- широкие возможности для управленческого учета и построения аналитической отчетности, поддержка много валютного учета;

- решение задач планирования, бюджетирования и финансового анализа;

- расчет зарплаты и управление персоналом;

- другие области применения.

1С может работать в нескольких режимах:

- 1С:Предприятие – основной режим работы пользователя, ввод данных, получение отчетов;

- конфигуратор – режим администрирования и изменения конфигурации;

- отладчик – режим отладки и замера производительности конфигурации;

- монитор – режим просмотра активных пользователей и журнала регистрации событий.

Логически всю систему можно разделить на две большие части, которые тесно взаимодействуют друг с другом: конфигурацию и платформу, которая управляет работой конфигурации.

Функционирование системы делится на два процесса:

– разработка (описание модели предметной области средствами системы);

– исполнение (обработка данных предметной области).

На этапе разработки система оперирует такими универсальными понятиями (объектами), как документ, журнал документов, справочник, реквизит, форма, регистр и другие. Совокупность этих понятий и определяет концепцию системы.

В процессе исполнения система уже оперирует конкретными понятиями, описанными на этапе конфигурирования (справочниками товаров и контрагентов, счетами, накладными и т. д.). При работе пользователя в режиме «1С:Предприятие 8.2» обработка информации выполняется как штатными средствами системы, так и с использованием алгоритмов, созданных на этапе конфигурирования.

Программа «1С:Сервисный центр» разработана для комплексной автоматизации сервисного центра специализирующегося на ремонте и обслуживании офисной техники. Данное отраслевое решение позволяет обрабатывать большие объемы информации, предоставить удобный интерфейс для работы с данными, позволяет пользователю, как просматривать необходимую информацию, так и, по мере необходимости, манипулировать ею.

«1С:Сервисный центр» автоматизирует учет всех операций, производящихся в сервисном центре:

- поступление товаров от контрагента на склад;
- перемещение товаров между складами;
- продажа товаров;
- оказание услуг;
- пересчет товаров (инвентаризация);
- заказ товаров от покупателя;
- поступление денежных средств от покупателей;
- оформление чеков продажи и сводного отчета по контрольно-кассовой машине;
- ведение учета выполненных услуг;
- ведение учета клиентов и поставщиков;
- учет доставок товаров и другие возможности.

Таким образом, «1С:Сервисный центр» позволяет автоматизировать процессы, связанные с работой сервиса. Внедрение данной подсистемы позволяет: повысить эффективность и снизить трудоемкость учета материалов; упростить работу сотрудникам.

## Литература

1. Радченко, М.Г. 1С:Предприятие 8.2. Практическое пособие разработчика / М.Г. Радченко, Е.Ю. Хрусталева. – Издательский дом «Вильямс», 2011. – 544 с.
2. Радченко, М.Г. 1С:Предприятие 8.2. Коротко о главном. Новые возможности версии 8.2 / М.Г. Радченко. – Официальное издание 1С, 2010. – 908 с.
3. Ажеронок, В.А. 1С:Предприятие 8.2. Руководство администратора / В.А. Ажеронок. – Издательский дом «Питер», 2012. – 789 с.
4. Хрусталева, Е.Ю. 1Сv8 Описание встроенного языка, учебный центр 1С / Е.Ю. Хрусталева. – Издательский дом «Вильямс», 2011. – 666 с.

**О.В. Чигринец (УО «Белорусский торгово-экономический университет потребительской кооперации», г. Гомель)**

Науч. рук. **Т.А. Заяц**, ст. преподаватель

### **ИСПОЛЬЗОВАНИЕ CASE-СРЕДСТВ В РАЗРАБОТКЕ АВТОМАТИЗИРОВАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ**

Разработка и исследование современных информационных систем является сложной задачей, решение которой требует применения специальных методик и инструментов.

В настоящее время создание крупных проектов практически невозможно без использования средств автоматизации проектирования (CASE-средств). CASE-средства (*Computer-Aided Software Engineering*) – это инструменты, которые позволяют автоматизировать процесс разработки информационной системы и программного обеспечения на всех его этапах.

Многие современные CASE-средства предоставляют возможности для моделирования практически всех предметных областей деятельности организаций. В составе этих средств существуют инструменты для описания моделей бизнес-процессов с помощью различных диаграмм, схем, графов и таблиц и инструменты для проектирования данных. В качестве примера можно привести инструментальное средство функционального моделирования бизнес-процессов и анализа деятельности организации *AllFusion Process Modeler 7* (в более ранних версиях *Business Process Modeler for Windows – BPWin*) и инструментальное средство автоматизации проектирования баз данных *AllFusion*

ERwin Data Modeler (краткое название *ERWin*), разработанные компанией Computer Associates (США).

В BPWin создается контекстная диаграмма, состоящая из одного блока, представляющего собой главный бизнес-процесс моделируемой системы (рисунок 1). Затем выполняется разбиение контекстной диаграммы на более мелкие компоненты (функциональная декомпозиция). Диаграммы, описывающие каждый компонент и их взаимодействие, называются *диаграммами декомпозиции*.

После декомпозиции контекстной диаграммы проводится декомпозиция каждого компонента системы на более мелкие компоненты. Процесс декомпозиции диаграмм повторяется до достижения нужного уровня детализации описания.

Декомпозиция позволяет постепенно и структурировано представлять модель системы в виде иерархической (древовидной) структуры отдельных диаграмм.

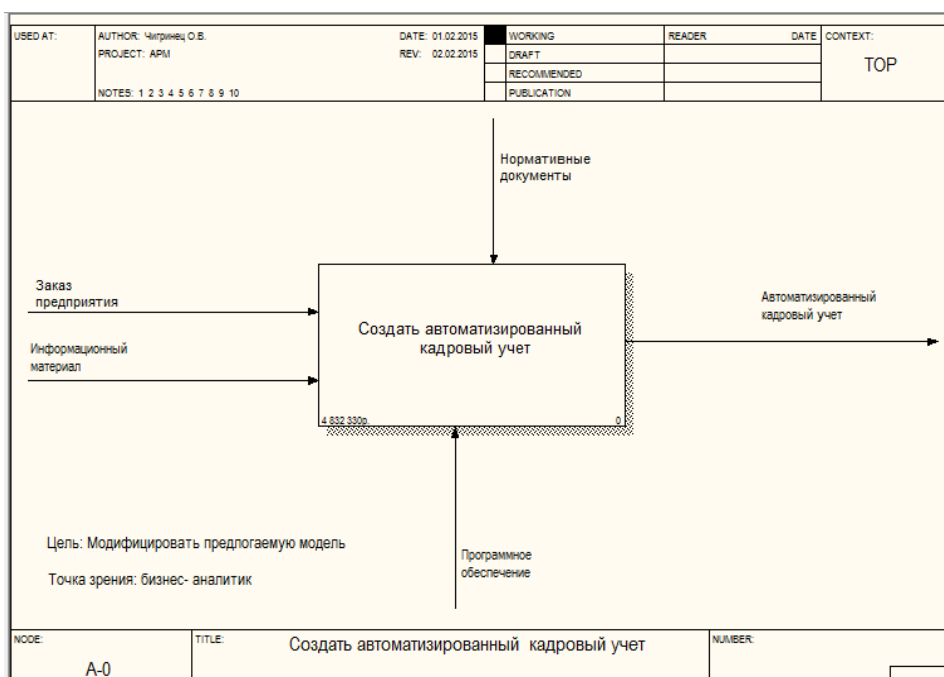


Рисунок 1 – Контекстная диаграмма

На рисунке 1 представлена контекстная диаграмма создания автоматизированной системы «Кадры». На рисунке 2 представлен пример диаграммы декомпозиции контекстной диаграммы, а именно детализация процесса создания АИС «Кадры».

Пакет ERWin является инструментальным средством, позволяющим автоматизировать процесс проектирования реляционных баз данных. В нем с помощью графических языков можно отобразить предметную область (построить концептуальную модель) и затем

осуществить автоматический переход от концептуальной модели к модели данных в среде выбранной целевой СУБД.

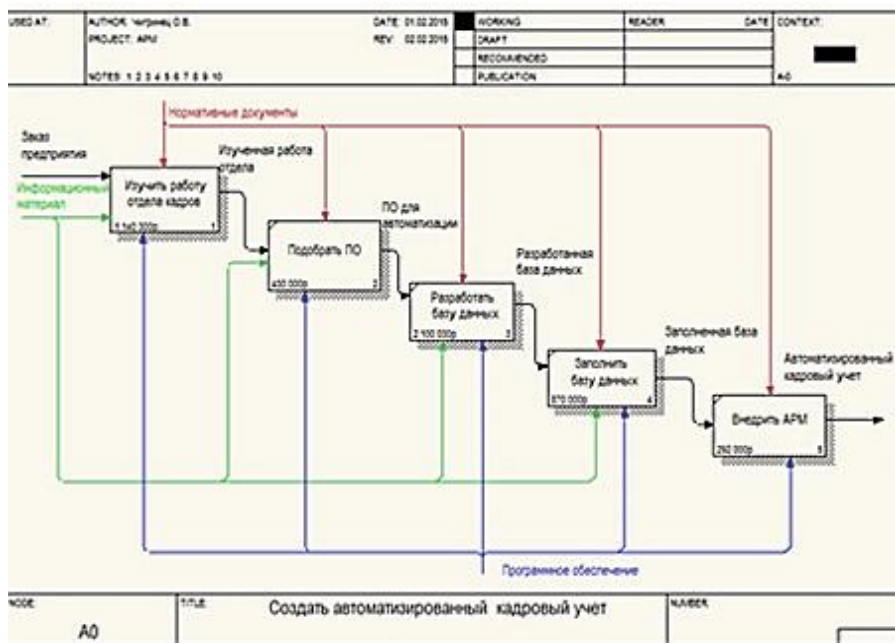


Рисунок 2 – Декомпозиция контекстной диаграммы

Использование AllFusion ERwin Data Modeler (ERWin) позволяет автоматизировать процесс проектирования реляционных баз данных. Для этого достаточно создать графическую E-R модель (объект-отношение), удовлетворяющую всем требованиям к данным и ввести бизнес-правила для создания логической модели, которая отображает все элементы, атрибуты, отношения и группировки. На рисунке 3 представлена логическая модель базы данных «Кадры», построенная в системе ERwin.

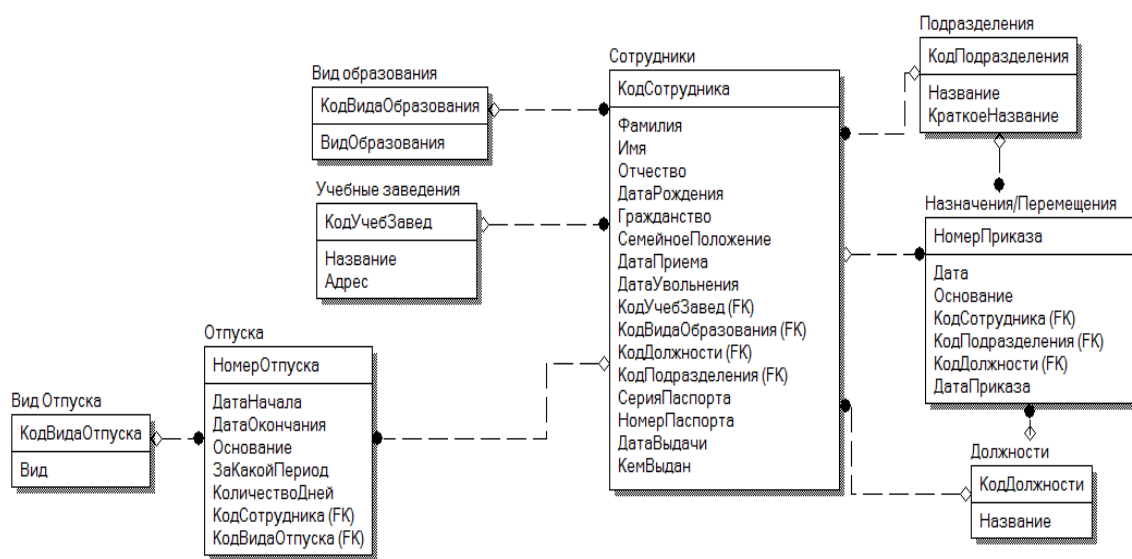


Рисунок 3 – Логическая модель базы данных «Кадры», построенная в системе ERwin



**Д.В. Чирик (ГГУ им. Ф. Скорины, Гомель)**  
Науч. рук. **Е.А. Левчук**, канд. техн. наук, доцент

## **АНАЛИЗ ЛОКАЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ ОВД РЕЧИЦКОГО РАЙИСПОЛКОМА**

Локальная вычислительная сеть Речицкого ОВД охватывает три этажа главного административного здания, два этажа второго административного здания, здание МРЭО ГАИ и здание отделения по гражданству и миграции. Кабельная система, сети построена на основе кабеля UTP категории 5е, в главном административном здании для связи с интрасетью применен волоконно-оптический кабель, что позволяет передавать информацию на высоких скоростях, топология комбинированная.

Локальная вычислительная сеть главного административного здания Речицкого ОВД содержит 6 коммутаторов компании HP. Серверное оборудование применено компании HP. Доступ в глобальную сеть реализован по средствам оборудования предоставленного провайдером «Белтелеком», на обособленные компьютеры не подключенные к корпоративной сети. Связь с остальными зданиями отдела организована по средствам ADSL модемов. В административном здании № 2, здании МРЭО ГАИ и здании отделения по гражданству и миграции локальная сеть организована в основном на оборудовании D-Link, так как количество подключенного оборудования изначально было не велико и нагрузка на этот сегмент была очень мала.

Реалии нашей жизни таковы, что с течением времени что-то развивается и модернизируется, а что-то деградирует и отстает в развитии. Эта участь не обошла стороной как сферу информационных технологий в целом, так и локальные вычислительные сети в частности. Исходя из вышеописанного, можно сделать вывод, что только в главном административном здании ОВД Речицкого райисполкома существует развитая, масштабируемая локально-вычислительная сеть с высокой степенью защищенности, а остальные ее сегменты находятся в эксплуатации уже довольно продолжительное время и нуждаются в модернизации.

**Д.В. Чирик (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **В.Д. Левчук**, канд. техн. наук, доцент

## **РАЗРАБОТКА ПРОЕКТА МОДЕРНИЗАЦИИ ЛОКАЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ ОТДЕЛА ВНУТРЕННИХ ДЕЛ РЕЧИЦКОГО РАЙИСПОЛКОМА**

Неизбежным этапом развития компьютерной техники было

объединение в одно логическое пространство, с целью обмена информацией. Компьютер организации не подключенный к корпоративной локальной вычислительной сети, по сути, является печатной машинкой. На подключенных же к сети компьютерах можно совместно использовать периферийные устройства, сервера, доступ в интернет и многое другое.

Локальная сеть ОВД Речицкого райисполкома охватывает три этажа главного административного здания (административное здание № 1), два этажа административного здания № 2, а так же здания МРЭО ГАИ и отделения по гражданству и миграции. Кабельная система сети имеет комбинированную топологию и построена на основе кабеля UTP категории 5е. Внешний канал от главного административного здания волоконно-оптический, что позволяет передавать большие объемы информации на высоких скоростях. Однако, связь между всеми зданиями осуществляется по технологии DSL (стандарт SHDSL), что на сегодняшний момент является узким местом.

Доступ в глобальную сеть реализован согласно требований политики безопасности, то есть на отдельные ПК, не подключенные к корпоративной сети. Основными задачами локальной вычислительной сети Речицкого РОВД являются:

- передача данных с городских камер наружного наблюдения;
- передача данных с камер наружного и внутреннего наблюдения всех административных зданий Речицкого РОВД и его подразделений;
- запись многоканального телефона специальной службы 102;
- обеспечение связи с серверами различных АРМ'ов по технологии клиент-сервер;
- совместное использование файловых серверов;
- совместное использование периферийных устройств;
- функционирование корпоративной почты.

Учитывая расположение зданий всех подразделений Речицкого РОВД и функциональные задачи, возлагаемые на ЛВС, можно заключить, что данная архитектура не является оптимальной и некоторые сегменты сети нуждаются в модернизации.

Локальная вычислительная сеть главного административного здания ОВД Речицкого райисполкома содержит 6 коммутаторов компании HP. Каналы коммутаторов имеют резервные линии. Для обеспечения большей безопасности применяется статическая IP-адресация. В качестве серверного оборудования были выбраны серверы компании HP.

В остальных зданиях подразделений ОВД Речицкого райисполкома используется оборудование компании D-Link – в основном это неуправляемые коммутаторы с максимальным числом портов не превышающим 12. Выбор такого оборудования при проектировании локальной вычислительной сети около 10 лет назад был оправдан тем, что

планируемое количество подключаемого оборудования будет не велико, можно сказать даже штучно. Основной задачей, возлагаемой на ЛВС в то время было обеспечение связи оператора и АРМ с одной стороны и сервера с другой. Следовательно, нагрузка на сеть была не велика.

Однако, время идет, развиваются технологии и растут запросы и на данном этапе локальная вычислительная сеть ОВД Речицкого райисполкома уже не может в полной удовлетворить все запросы.

Все здания подразделений ОВД Речицкого райисполкома достаточно удалены друг от друга и в настоящее время связь между ними осуществляется по средствам DSL модемов по стандарту SHDSL, максимальная скорость которого по сегодняшним меркам не велика. Коммутаторы, за исключением тех, что используются в главном административном здании, не имеют свободных портов и не имеют модульную конструкцию, так же они не являются управляемыми.

Таким образом, для решения создавшейся проблемы решено установить 3 коммутатора (по одному на каждое здание) Cisco WS-C2960-24-S, что обеспечит возможность контроля и разграничения трафика, а так же хорошую масштабируемость сети. Для обеспечения высокой пропускной способности между сегментами сети вертикальную подсистему ЛВС решено реорганизовать по технологии GPON. Для объединения подсетей через VPN, понадобится 3 оптоволоконных маршрутизатора, коммутация подсетей будут производиться непосредственно на мощностях Белтелекома.

Исходя из вышеописанного, можно сделать вывод, что на данный момент локальная вычислительная сеть ОВД Речицкого райисполкома не в полной мере удовлетворяет возложенным на нее задачам. Только хорошо спроектированная и продуманная локальная сеть с выбором оптимального по цене и функционалу оборудования позволит в будущем избежать больших затрат времени и средств.

**Г.А. Шангытбаева (КазНТУ им. К. Сатпаева, Алматы, Казахстан)**  
Науч. рук. **Б.С. Ахметов**, д-р техн. наук, профессор

## **ВЛИЯНИЕ АТАК НА ОТКАЗ В ОБСЛУЖИВАНИИ НА КОМПЬЮТЕРНЫЕ СЕТИ**

Среди многочисленных атак злоумышленников на компьютерные сети наиболее распространены прерывания и искажения пакетного трафика. Самыми разрушительными атаками на сегодняшнее время являются атаки, направленные на отказ в обслуживании законных услуг.

Согласно отчету данных компании Prolexis, лидера мирового рынка защиты от атак на отказ от обслуживания, средняя мощность атаки составляет 49 Гб / с, 17 % атак имеют мощность более 60 Гб / с. Относительный прирост среднего значения мощности составляет 925 % или 1655 % пакетов в секунду [1]. Атаки низкой мощности являются целевыми уровня HTTP Flood или SYN Flood. Максимальная мощность используется для защищенных ресурсов компьютерной сети. Атака уничтожает не только сайт, но и все хост и сеть провайдера. Таких атак с каждым годом становится все больше.

На практике для минимизации потерь от атаки используют методы фильтрации UDP трафика. Такой метод эффективен для кратковременного защиты автономных систем глобальной сети при наличии современных средств фильтрации в пограничной части сети и управлении высококвалифицированными специалистами. Такие программно-аппаратные комплексы являются дорогостоящими и не проявляют инцидентов атаки. На рисунке 1. показано загрузки средства фильтрации при реализации реальной атаки на отказ в обслуживании [2].

Анализ показал, что существующие средства защиты не справляются с управлением трафика путем фильтрации. Поэтому возникает острая необходимость в создании новых методов и средств защиты архитектур типа клиент-сервер от атак на отказ в обслуживании.

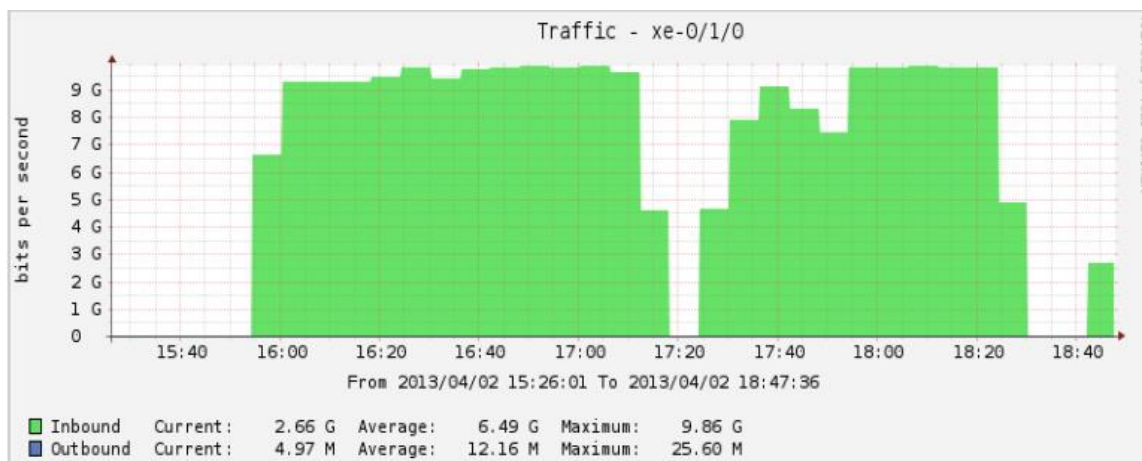


Рисунок 1 – Работа UDP – фильтра при DDoS-атаки

Проанализировав подходы к методам отслеживания атак, обнаружено, что для задачи трассировки IP-адресов перспективным методом является вероятностное маркировки пакетов.

Этот подход целесообразно применять во время атаки или после проведения атаки. В методе вероятностного маркировки пакетов не генерируется дополнительный сетевой трафик, а хранится информация о маршрутизаторах увеличивается размер пакета.

Для отслеживания IP-адреса инициатора атаки на отказ в обслуживании целесообразно использовать метод вероятностного маркировки пакетов. Известно, что в данном методе каждый маршрутизатор вероятностно вписывает свою локальную информацию о пути следования пакета до конечного узла. Таким образом, пользователь с большой вероятностью может восстановить полный путь прохождения пакетов, проверяя маркировку. В алгоритме вероятностного маркировки пакетов каждый маршрутизатор случайно определяет вероятность маркировки, который перезаписывает информацию в поле маркировки, уничтожая маркировки пройденных маршрутизаторов.

### Литература

1. Ioannidis J. Implementing pushback: Router-based defense against DDoS attacks / J. Ioannidis, S.M. Bellovin // In proceedings of network and distributed system security symposium : The Internet Society, 2002 : thesis. – 2002. – P. 26–38.
2. Burch H. Tracing anonymous packets to their approximate source / H. Burch, B. Cheswick // In Usenix LISA Conference : thesis. – New Orleans, 2000. – P. 313–322.

**Г.А. Шангытбаева (КазНТУ им. К. Сатпаева, Алматы, Казахстан)**  
Науч. рук. **Б.С. Ахметов**, д-р техн. наук, профессор

### РАЗРАБОТКА СРЕДСТВ ЗАЩИТЫ ИНФОКОММУНИКАЦИЙ НА ОСНОВЕ ПЛИС

Программируемые логические интегральные схемы (ПЛИС) нашли широкое применение для решения ресурсомисных задач использования надежных и безопасных высокоскоростных информационных коммуникаций. Так, на сегодняшний день, ПЛИС компании Xilinx (США) имеют как программные, так и аппаратные решения защиты передачи информации, работа которых основана на использованные пассивных и активных методов.

Среди семейств ПЛИС компании Xilinx высокий уровень защиты реализовано в семействе UltraScale (7 пассивных и 10 активных методов), несколько ниже – Zynq, ПЛИС Xilinx 7-й серии, Virtex-6 (соответственно 5 и 8), еще ниже – в ПЛИС семейства Spartan-6 (4 и 4) [1].

Решение с использованием ПЛИС характеризуются гибкостью процесса разработки, что обусловлено простотой внесения изменений при корректировке заказчиком функциональной части, а благодаря

реализации проекта на одной из языков описания оборудования существенно упрощается проверка соответствия техническому заданию разработанного средства.

Преимуществом замены специализированных микросхем кристаллами ПЛИС является существенное уменьшение количества и длины проводников, уровня электромагнитного и теплового излучения, что в свою очередь уменьшает утечки информации через каналы побочных электромагнитных излучений и наводок.

Защищенность программной и аппаратной частей от несанкционированного доступа к данным позволяет разрабатывать функциональные узлы тестирования и настройки средств защиты инфокоммуникаций на основе ПЛИС. Принцип их работы заключается в подаче псевдослучайной последовательности на вход системы, сопоставимые результирующего сигнала на выходе из эталонным. Псевдослучайную последовательность получают на основе рекуррентных линий задержки, которые проецируются на аппаратуру как триггеры смещения. Обратная связь обеспечивается операцией сложения по модулю два, которое реализуется элементом XOR.

Проектирование средств защиты инфокоммуникаций заключается в выполнении следующих этапов:

- 1) разработка аналитической модели работы;
- 2) построение алгоритма работы и оптимизация по критериям временной и пространственной сложностей;
- 3) проектирование алгоритма на аппаратуру путем его описания на языке VHDL или Verilog, разработки IP-ядер;
- 4) функциональное и временное моделирование, топологическое проектирование логического и коммуникационной среды;
- 5) размещение проекта на кристалле и верификация проекта с целью выявления влияния реальных задержек распространения сигналов;
- 6) завершающий этап – конфигурирование ПЛИС.

Этапы 1–5 являются итерационными и требуют верификации проекта. В случае сложности проекта применяют общесистемный уровень на основе подуровней спецификации системы, сообщений и передач. Такой подход позволяет сразу проектировать как программную, так и аппаратную части системы, согласовывать их между собой начиная с начальных этапов, подготовить всю экосистему до момента получения прототипа.

### **Литература**

1. Design Security: [Электронный ресурс]. – Режим доступа: <http://www.xilinx.com/products/technology/design-security/index.htm>
2. Маршрут проектирования «Систем-на-кристалле» (СнК): [Электронный ресурс]. – Режим доступа: <http://www.ipmce.ru/custom/ekb/mp/>

**А.О. Шараев (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **А.В. Воруев**, канд. техн. наук, доцент

## **ТЕНДЕНЦИИ РАЗВИТИЯ СЕТЕЙ МАСШТАБА ГОРОДА И ПРЕДОСТАВЛЯЕМЫХ ИМИ СЕРВИСОВ**

Нарастающая конкуренция на рынке местных линий связи между операторами заставляет их оптимизировать свои сети в целях предоставления наиболее выгодных, с точки зрения стоимости, услуг.

В настоящее время в мире активно развиваются сети N-Play (передача данных, голоса и видео с использованием сетей IP, другие услуги), наблюдается масштабное развитие широкополосного доступа с применением технологий Metro Ethernet, которые приобрели популярность при создании городских сетей в силу своей масштабируемости, универсальности и удовлетворения повышенных требований, предъявляемых новыми услугами к полосе пропускания, надежности, качеству обслуживания сети.

Проведение оптоволоконного кабеля в жилые дома, многоквартирные здания и помещения с организациями типа малого офиса или домашнего офиса становится жизнеспособной деловой возможностью в плотно заселенных городских районах.

Потенциальный спрос на услуги высокоскоростного доступа в Интернет очень большой и продолжает расти. Это нужно и интересно всем категориям пользователей от частных домохозяйств до предприятий с распределенной инфраструктурой и малого бизнеса. Используя инфраструктуру «широкополосного доступа» предприятия решают наиболее важные задачи отношений с клиентами и управления, а частные пользователи получают удовольствие и делают свою жизнь удобней.

Скорость, экономичность и выгодность с точки зрения затрат, а также простота, легкость использования и известность Ethernet являются большим плюсом в новом подходе к построению городских сетей. Городские сети (Metropolitan Area Networks) Ethernet-доступа, обеспечивающие настоящий широкополосный доступ, становятся все более жизнеспособным решением, обеспечивающим большую пропускную способность по конкурентным ценам. Но, что наиболее важно, это открывает возможность для предоставления прибыльных дополнительных услуг.

Metro Ethernet как среда реализации коммуникационных сервисов представляет собой технологическую базу для доставки услуг. Это понятие охватывает оптические и другие сети Ethernet масштаба города. Решения Metro Ethernet становятся основной сервисной архитектурой в городах.

Рассмотрим особенности наиболее популярных Ethernet-услуг, востребованных операторами связи:

– Выделенный доступ в Интернет. Поскольку пользователи заинтересованы в получении высокоскоростного подключения к Интернету, услуга EVC «точка–точка» идеально обеспечивает подключение пользовательского сайта к ближайшей точке присутствия (POP) оператора Интернет (ISP).

В простейшем сценарии на сайте пользователя могут использоваться непомеченные сервисные кадры. Для подключения к двум и более операторам пользователь может задействовать Border Gateway Protocol (BGP). В этом случае он должен использовать отдельные E-Line к каждому ISP.

Оператору рекомендовано использовать мультиплексирование услуг пользователей на одном высокоскоростном Ethernet UNI.

– Расширение ЛВС. Представим себе, что есть потребитель, имеющий множество сайтов в одном населенном пункте (metro area), который желает соединить их между собой на высокой скорости, чтобы эти сайты выглядели как одна ЛВС с производительностью, эквивалентной офисной ЛВС, и при этом был обеспечен доступ к серверам и хранилищам данных предприятия.

Это наиболее характерный пример услуги «Расширения (extension) ЛВС», обеспечивающей подключение пользователей ЛВС без какой-либо промежуточной маршрутизации между различными UNI (сайтами). В некоторых случаях это проще и дешевле, чем маршрутизация, однако для очень больших сетей могут возникать проблемы масштабируемости.

Услуга «Расширение ЛВС» предполагает соединение коммутатор–коммутатор, а это требует большей прозрачности от сети оператора, чем для услуги «Выделенный доступ в Интернет». Например, пользователь может захотеть использовать Spanning Tree Protocol (STP) сквозь соединенные сайты, требуя при этом от оператора поддержки туннелей Bridge Packet Data Unit (BPDU). Если в сети пользователя используются VLAN, разделяющий трафик различных департаментов, пользователю также понадобится сделать представление VLAN на множестве сайтов, требуя для этого поддержку переноса пользовательских CE-VLAN-меток через MEN-соединения. Каждый интерфейс должен поддерживать сохранение CE-VLAN ID и CE-VLAN CoS – другими словами, метка пользовательского VLAN и бит 802.1p не должны преобразовываться в MEN. В этом случае MEN выглядит как одиночный Ethernet-сегмент, в котором каждый сайт может быть участником любого VLAN. Преимущество такого подхода заключается в том, что



пользователь может конфигурировать CE-VLAN сквозь все четыре сайта без каких-либо координаций и взаимодействия с оператором.

– Интранет/экстранет L2 VPN. Ethernet-услуги могут быть хорошим вариантом для маршрутизируемых интранет-соединений между удаленными сайтами и экстранет-соединений для поставщиков, заказчиков и других бизнес-партнеров организации пользователя.

– Телевизионное вещание (Video Broadcast). На базе соединений типа E-Tree может быть организовано телевизионное вещание – передача видеосигналов от источника к множеству потребителей. В этом сценарии от сайта, в котором расположен передающий видеоцентр (video head-end), через UNI, являющимся корнем, организовано множество EVC к UNI пользователей (листикам). Соединение может поддерживать множество вещательных каналов, которые доставляют данные от корня к листикам, другими словами, от video head-end – к пользователям, в одном направлении. Подобный сценарий обладает преимуществами масштабирования не хуже, чем соединение типа E-Line.

Пользователь может получать также определенные каналы, на которые он подписан. Сигнализация, отвечающая за определение этих каналов для каждого отдельного пользователя, реализуется через стандартный multicast-протокол, например IGMP v3.

В случае, когда необходимо резервирование, может использоваться два корневых UNI, взаимодействие между которыми происходит с помощью протоколов резервирования, отвечающих за то, чтобы только от одного корня передавались данные через EVC к пользователям.

Используя соединения Metro Ethernet в качестве основы для услуг оператора, можно обеспечить поддержку значительного набора существующих приложений более легко, эффективно и выгодно с точки зрения затрат, чем с помощью других сетевых возможностей и стандартов.

Используя типовой Ethernet-интерфейс, пользователь может установить безопасное выделенное виртуальное Ethernet-соединение через метро-сеть или через WAN, объединить свои сайты, подключить бизнес-партнеров, поставщиков или клиентов, а также обеспечить соединение с Интернетом.

В дополнение, многие услуги, созданные на базе соединений Metro Ethernet, обладают уникальной гибкостью по управлению гарантированной пропускной способностью. Это позволяет пользователю не задумываться над тем, каким образом он сможет увеличить пропускную способность канала между своими сайтами, если этого требуют приложения.

Залогом настоящего и будущего развития услуг на базе Metro Ethernet является их простота, относительная дешевизна и доступность при условии реализации соответствующего архитектурного решения у оператора.

**И.С. Шестаков (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.С. Давыдов**, канд. техн. наук, доцент

## **FRONT-END И BACK-END РАЗРАБОТКА ПРИЛОЖЕНИЯ ПО УЧЕТУ ИСКЛЮЧИТЕЛЬНЫХ СИТУАЦИЙ В ИНФОРМАЦИОННОЙ СИСТЕМЕ**

Разработка сайтов по системе front-end и back-end подразумевает иерархическое разделение процесса создания ресурса на две части, на разработку пользовательского интерфейса – (*фронтэнда*) и его программно-административной части (*бэкэнда*).

Front-end разработка – это работа по созданию публичной части сайта, с которой непосредственно контактирует пользователь и функционала который обычно отображается на клиентской стороне. К аспектам фронтэнд разработки можно отнести создание дизайн-макета сайта, верстку сайтов и шаблонов для CMS, а также привязку к пользовательскому интерфейсу специальных скриптов, отвечающих за визуализацию и web-анимацию. Front-end составляющая часть сайта отвечает за вывод определенной информации пользователю ресурса.

Back-end разработка – это процесс программирования сайта и наполнения его функционалом. Создание ядра сайта, разработка платформы сайта, наполнение его основным функционалом и создание административной зоны – это и есть бэкэнд разработка. Back-end производит обработку пользовательской информации полученной из front-офиса, и возвращает front-end’у результат в понятной ему форме.

В приложении по учету исключительных ситуаций в информационной системе front-end частью являются jsp страницы, которые содержат в себе структуру приложения, формы для заполнения информации по исключительным ситуациям, формы для регистрации и редактирования пользователей, формы для создания и редактирования дополнительных параметров. Также к front-end относятся файлы стилей css.

Back-end часть в приложении написана на языке Java. К ней относятся: сервлеты, bean-классы, интерфейсы и другие классы приложения. Back-end логика отвечает за обработку данных введенных пользователем из front-end части.

## Литература

1 Habrahabr [Электронный ресурс]. – 2015. – Режим доступа: <http://habrahabr.ru/post/231855/>. – Дата доступа: 13.04.2015..

2 Front-end и Back-end разработка [Электронный ресурс]. – 2012. – Режим доступа: <http://konservs.com/it/web/front-and-back-end-61>. – Дата доступа: 13.04.2015.

**И.С. Шестаков (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.С. Давыдов**, канд. техн. наук, доцент

### **ИСПОЛЬЗОВАНИЕ JAVA BEANS ПРИ СОЗДАНИИ WEB-ПРИЛОЖЕНИЯ ПО УЧЕТУ ИСКЛЮЧИТЕЛЬНЫХ СИТУАЦИЙ В ИНФОРМАЦИОННОЙ СИСТЕМЕ**

JavaBeans – классы в языке Java, написанные по определённым правилам. Они используются для объединения нескольких объектов в один (англ. Bean – фасоль, кофейное зерно) для удобной передачи данных.

Спецификация Sun Microsystems определяет JavaBeans как повторно используемые программные компоненты, которыми можно управлять, используя графические конструкторы и средства IDE.

JavaBeans обеспечивают основу для многократно используемых, встраиваемых и модульных компонентов ПО. Компоненты JavaBeans могут принимать различные формы, но наиболее широко они применяются в элементах графического пользовательского интерфейса. Одна из целей создания JavaBeans – взаимодействие с похожими компонентными структурами. Например, Windows-программа, при наличии соответствующего моста или объекта-обёртки, может использовать компонент JavaBeans так, будто бы он является компонентом COM или ActiveX.

Чтобы класс мог работать как bean, он должен соответствовать определённым соглашениям об именах методов, конструкторе и поведении. Эти соглашения дают возможность создания инструментов, которые могут использовать, замещать и соединять JavaBeans.

Правила описания гласят:

1. Класс должен иметь конструктор без параметров, с модификатором доступа public. Такой конструктор позволяет инструментам создать объект без дополнительных сложностей с параметрами.

2. Свойства класса должны быть доступны через get, set и другие методы (так называемые методы доступа), которые должны подчиняться стандартному соглашению об именах. Это легко позволяет

инструментам автоматически определять и обновлять содержание bean'ов. Многие инструменты даже имеют специализированные редакторы для различных типов свойств.

3. Класс должен быть сериализуем. Это даёт возможность надёжно сохранять, хранить и восстанавливать состояние bean независимым от платформы и виртуальной машины способом.

Так как требования в основном изложены в виде соглашения, а не интерфейса, некоторые разработчики рассматривают JavaBeans, как Plain Old Java Objects, которые следуют определённым правилам именования.

В приложении по автоматизации учета исключительных ситуаций есть восемь bean-классов (Issue, Priority, Project, Resolution, Role, Status, Type, User). У каждого из этих классов есть собственный конструктор и конструктор по умолчанию. Для обеспечения доступа к данным у каждого класса есть get и set методы.

### Литература

1. Блинов, И.Н. Java 2: практическое руководство / И.Н. Блинов, В.С. Романчик. – Мн.: УниверсалПресс, 2005. – 318 с., ил.

2. Онлайн учебник по JSP [Электронный ресурс]. – 2015. – Режим доступа: <http://www.java2ee.ru/jsp/javaBean.html>. – Дата доступа: 13.04.2015.

3. Свободная энциклопедия Википедия [Электронный ресурс]. – 2013. – Режим доступа: <https://ru.wikipedia.org/wiki/JavaBeans>. – Дата доступа: 13.04.2015.

**В.А. Шинкарёва (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Леванцов**, ст. преподаватель

### СИСТЕМА УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ CUBRID

CUBRID является полноценной свободной системой управления базами данных с открытым исходным кодом хорошо оптимизированная для веб-приложений, особенно когда сложные бизнес процессы оперируют большими данными и вырабатывают огромное количество параллельных запросов.

Предоставляя уникальные оптимизированные функции, CUBRID позволяет обрабатывать гораздо больше параллельных запросов за более короткое время. С помощью CUBRID компании смогут воспользоваться преимуществами более высокой производительности

и стабильности, масштабируемости и высокой доступности, требуемые организациями для обеспечения круглосуточного сервиса для своих ценных клиентов.

СУБД продавалась и пользовалась очень большим спросом еще задолго до того, как появился MySQL, и даже сам CUBRID. Она была одной из первых с объектно-ориентированной архитектурой, которая широко используется и в наши дни в игровой и мультимедийной индустриях.

Официально годом начала разработки CUBRID определяется 2006 г., когда корпорация NHN (ведущий поставщик ИТ-услуг в Корее), объединила в команду из 40 человек своих главных архитекторов и программистов и организовала проект CUBRID. Компания NHN решает создать объектно-реляционную СУБД, которая предоставляла бы все преимущества и ООСУБД, которая так часто используется в онлайн играх и мультимедийных службах, и РСУБД, которая стала самым популярным решением для всех других отраслей.

Первый публичный релиз был анонсирован в октябре 2008 г. На сегодняшний день было объявлено 15 релизов в интервале от двух до шести месяцев. Это является одним из самых активных и продвинутых проектов с открытым исходным кодом на условиях GPL v2.0 и выше. Основная цель проекта CUBRID является разработка реляционной СУБД, оптимизированной для веб-услуг.

Популярность CUBRID на внутреннем рынке выросла насколько быстро, что в течение года уже несколько тысяч пользователей начали разрабатывать и адаптировать разные приложения для работы с СУБД CUBRID, как LACP (Linux, Apache, CUBRID, PHP/Perl/Python) и LnCP (Linux, nginx, CUBRID, PHP/Perl/Python) стэки, Windows установщики, а также известные CMS (WordPress, phpBB, Joomla).

На сегодняшний день СУБД CUBRID разрабатывается для двух основных операционных систем. Это Linux и Windows, для которых доступны серверная часть CUBRID, все клиентские приложения и интерфейсы программирования. Для Mac OS X на данный момент доступны только клиентские приложения, с помощью которых можно полноценно работать с удаленными серверами CUBRID. Однако разработки основной серверной части CUBRID для Mac OS в планах еще нет.

Серверная часть CUBRID разрабатывается на языке программирования C/C++ и распространяется под лицензией GPL версии 2.0 или выше. Клиентские приложения разработаны на разных языках и обычно распространяются под лицензией BSD. Основные инструменты для администрирования базами данных CUBRID Manager, Query Browser и Migration Toolkit написаны на языке Java. А интерфейсы программирования разрабатываются на C.

В реализации реляционной части CUBRID ссылается на открытый стандарт SQL 1992 г. Многие СУБД его поддерживают, но каждая из них реализовывает его по-разному.

Например, системные таблицы, которые хранят метаданные о всех существующих или об определенной базе. Для этого в MySQL, MSSQL и некоторых других СУБД существуют отдельные системные базы, например, INFORMATION\_SCHEMA, которые доступны для прямого редактирования только самой системе. Это удобно, за исключением того, что при переносе баз данных на другой сервер, системные базы/таблицы на новом сервере (и на старом тоже) должны быть обновлены. Обычно это происходит автоматически при восстановлении баз данных, что требует дополнительных ресурсов, особенно если в базе сотни или тысячи таблиц. Страшнее всего – это когда системные таблицы не обновляются вообще или доступ к ним меняется. В данном случае требуется прямое вмешательство администратора или изменение клиентских приложений.

В CUBRID системные таблицы реализованы немного по-другому. Каждая база в CUBRID имеет свои системные классы-каталоги и виртуальные классы-каталоги, которые держат в себе данные об этой базе, включая все индексы, столбцы, пользователи, триггеры.

В CUBRID есть и таблицы, и столбцы, и процедуры. Доступ к данным в CUBRID возможен разными способами. Для доступа к таблице, можно использовать таблицы (реляционный подход) и классы (объектный подход). Для доступа к строкам, можно использовать так строки (реляционный подход), так и экземпляры классов (объектный подход). Столбцы, либо атрибуты. Процедуры или методы. Таким образом можно использовать обычный SQL (`SELECT index_name FROM db_index`)), чтобы извлечь, например, все имена индексов, которые используются во всей базе. Нет необходимости ссылаться на внешнюю базу. Можно также уточнить, чтобы индексы были только первичные, либо обратные или уникальные, либо только внешние ключи.

В CUBRID реализован ACID, таким образом есть полная поддержка транзакций. В CUBRID можно производить разделение, репликацию, сжатие, проверку и восстановление данных. Также возможно делать горячий/онлайн бэкап, создавать обновляемые представления, триггеры, иерархические и вложенные запросы. В CUBRID нет ограничений на размеры базы данных, количество таблиц или строк, и даже на размеры определенных типов данных, как BLOB и CLOB. В нем есть курсоры, а также встроенные функции-счетчики. CUBRID также позволяет кэшировать и планировать запросы. Есть много способов для моментальных оптимизаций запросов с помощью SQL подсказок.

На сегодняшний день разработано очень большое количество функций в CUBRID, множество из которых полностью совместимы с другими PCУБД, как MySQL или MSSQL, и в то же время есть очень много уникальных особенностей. Для удобства пользователей предоставлена максимальная совместимость с MySQL, чтобы при переходе на CUBRID пользователи могли без труда адаптировать свои приложения.

CUBRID обладает неплохим инструментарием для администрирования, хорошей реализацией высокой доступности (failover, обновление СУБД и ОС без простоя), резервного копирования (горячие бэкапы, компрессия). Готовы и инструменты для миграции: Scriptella и Apache DdlUtils. В качестве хранилища CUBRID уже могут использовать MediaWiki, phpBB, Wordpress.

К минусам относятся: небольшое сообщество разработчиков и пользователей, отсутствие поддержки Solaris, Mac OS X и FreeBSD, а также некоторые особенности диалекта SQL.

В целом, CUBRID – это полнофункциональная система управления базами данных, которая может предоставить бесперебойную работу с данными при очень высоких нагрузках. К примеру, в NHN СУБД CUBRID используется на серверах поисковой службы NAVER, которая принимает запросы от более, чем 17 миллионов уникальных пользователей в день. CUBRID используется в системе мониторинга результатов поиска на сайте NAVER.com и непосредственно отвечает за хранение данных о качестве результатов. Для улучшения релевантности результатов запросов и борьбы со спам-сайтами необходимо вести запись ключевых слов, которые используются в поиске, и ассоциировать каждую из них со всеми Веб страницами, которые уже проиндексированы поисковым сервером. Миллионы записей то вводятся, то обновляются, и конечно же, извлекаются из базы, и CUBRID справляется с этим безупречно.

**В.А. Шинкарёва (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **В.Н. Кулинченко**, ст. преподаватель

## **СРАВНИТЕЛЬНАЯ ХАРАКТЕРИСТИКА АНОНИМНЫХ СЕТЕЙ I2P И TOR**

I2P (англ. «Invisible Internet Project») – открытое ПО, созданное для организации сверхустойчивой анонимной, оверлейной, зашифрованной сети и применимое для веб-серфинга, анонимного хостинга, систем обмена сообщениями, ведения блогов, а также для файлообмена,

электронной почты, VoIP и многого другого. Адреса сайтов в сети I2P находятся в псевдо-доменном пространстве .i2p.

I2P – это сеть внутри сети. Она предназначена для защиты передачи данных от внешнего наблюдения и надзора. Внутри сети I2P работает собственный каталог сайтов, электронные библиотеки, торрент-трекеры. I2P доступна на компьютерах, встраиваемых системах и Android устройствах. Кроме того, существуют гейты для доступа в сеть I2P непосредственно из Интернета. Для доступа в I2P необходимо установить на компьютере программу-маршрутизатор, которая (де)шифрует, (раз)сжимает трафик и направляет его пирам в I2P. Для работы с внутрисетевыми сайтами необходимо настроить браузер для направления HTTP-пакетов роутеру. Для обращения к внешнему интернету через I2P необходимо использовать прокси-серверы изнутри I2P.

Весь трафик в сети шифруется от отправителя до получателя. При пересылке сообщения используется четыре уровня шифрования, перед шифрованием в каждый сетевой пакет автоматически добавляется небольшое случайное количество случайных байт, чтобы еще больше обезличить передаваемую информацию. В качестве адресов сети используются криптографические идентификаторы. IP адреса в сети I2P не используются, поэтому определить истинный адрес какого-либо узла в сети не представляется возможным. Все передаваемые сетевые пакеты имеют свойство расходиться по нескольким разным туннелям, что делает бессмысленными попытки прослушать и проанализировать с помощью сниффера проходящий поток данных. Также происходит периодическая смена (каждые 10 минут) уже созданных туннелей на новые, с новыми цифровыми подписями и ключами шифрования. В сети I2P все пакеты зашифровываются на стороне отправителя и расшифровываются только на стороне получателя. При этом никто из промежуточных участников обмена не имеет возможности перехватить расшифрованные данные, и никто из участников не знает, кто на самом деле отправитель и кто получатель.

Проект I2P был создан для тех, кто заинтересован в нецензурируемом, анонимном средстве общения и распространения информации. Все аспекты сети доступны в виде исходного кода и бесплатны.

Tor (англ. – The Onion Router) – свободное ПО для реализации второго поколения так называемой «луковой маршрутизации». Это система, позволяющая устанавливать анонимное сетевое соединение, защищенное от прослушивания. Рассматривается как анонимная сеть, предоставляющая передачу данных в зашифрованном виде.

Tor – это не только ПО, но и распределенная система серверов, между которыми трафик проходит в зашифрованном виде. На последнем сервере-ноде в цепочке передаваемые данные проходят процедуру



расшифровки и передаются целевому серверу в открытом виде. Кроме того, через заданный интервал времени (около 10 минут) происходит периодическая смена цепочки. При таком подходе вскрыть канал можно только при взломе всех серверов цепочки, что практически нереально, так как они располагаются в разных странах, а сама цепочка постоянно меняется.

Шифрование производится следующим образом. Перед отправлением пакет последовательно шифруется тремя ключами: сначала для третьей ноды, потом для второй и, в конце концов, для первой. Когда первая нода получает пакет, она расшифровывает «верхний» слой шифра и узнает, куда отправить пакет дальше. Второй и третий серверы поступают аналогичным образом.

Tor может работать не только с веб-браузерами, но и со многими существующими приложениями на основе протокола TCP. Приложения для работы в Сети, в простейшем случае это браузер, необходимо еще настроить на работу с Tor.

Система Tor позволяет скрывать от провайдера конечные адреса, тем самым, прорывая возможную блокаду доступа к заблокированным им сетевым ресурсам. Также система Tor надежно скрывает от целевых ресурсов адрес отправителя.

Однако Tor допускает перехват самого содержимого сообщений (без выявления отправителя) из-за необходимости их расшифровки на выходном узле. Впрочем, для такого перехвата нужно поставить на выходных узлах анализатор трафика, что не всегда просто сделать. Особенно, если учесть, что выходные узлы постоянно меняются.

Еще одним достоинством Tor является то, что это свободное ПО. То есть распространение его полностью бесплатно и с открытым исходным кодом. Проект Tor является некоммерческой организацией, поддерживающей и развивающей ПО Tor.

#### *Преимущества Tor перед I2P:*

- большая пользовательская база;
- разрешен ряд проблем масштабирования, к которым I2P еще не подошел;
- имеет значительное финансирование;
- большое количество разработчиков;
- более устойчивый к блокировкам на уровне государств, благодаря транспорту поверх TLS и мостам;
- достаточно большая сеть, чтобы адаптироваться к блокировке и попыткам DOS;
- разработан и оптимизирован для выхода трафика, с большим числом выходных узлов;

- более качественная документация, есть исследования и спецификации;
- эффективен в использовании памяти;
- клиенты Tor работают с небольшими затратами на протоколе;
- централизованный контроль уменьшает сложность каждого узла и может эффективно работать с атаками Sybil;
- набор высокопроизводительных узлов обеспечивает высокую производительность и меньшие задержки;
- реализация на С.

#### *Преимущества I2P перед Tor*

- полностью распределенная и самоорганизующаяся сеть;
- разработан и оптимизирован для работы скрытых сервисов;
- пиры выбираются на основе постоянного профайлинга и замеров по производительности;
- справочные сервера постоянно меняются и не обладают доверием;
- обеспечивает пиринговые сервисы;
- коммутрует пакеты, нежели соединения;
- явная прозрачная балансировка сообщений по многим пирам, в отличие от использования одного пути;
- надежность и отказоустойчивость за счет поддержания нескольких параллельных туннелей и обеспечения ротации туннелей;
- защита против детектирования клиентской активности;
- в I2P реализованы короткоживущие туннели, в отличие от цепочек в Tor, которые как правило живут долго;
- API I2P спроектированы под анонимность и безопасность;
- все пиры участвуют в маршрутизации трафика для других;
- затраты на работу в полном режиме довольно низкие;
- встроенный механизм автоматических обновлений;
- используется как TCP, так и UDP транспорт;
- реализация на Java.

**А.Н. Штукарь (УО «ГГТУ имени П.О. Сухого», Гомель)**

Науч. рук. **В.С. Мурашко**, ст. преподаватель

### **АВТОМАТИЗИРОВАННЫЙ РАСЧЕТ РАЦИОНАЛЬНЫХ РЕЖИМОВ РЕЗАНИЯ НА ПЛОСКОШЛИФОВАЛЬНЫХ СТАНКАХ**

Целью данной работы является автоматизация определения рациональных режимов обработки на плоскошлифовальных станках.

Плоское шлифование – это высокопроизводительный метод обработки незакаленных и закаленных деталей различного назначения. Плоское шлифование в ряде случаев применяют вместо чистового строгания, чистового фрезерования и шабрения. Плоские поверхности можно шлифовать периферией и торцом круга. При шлифовании на круглом столе шлифуются некрупные заготовки (торцы колец, втулок, шатунов) в серийном и массовом производстве.

Необходимость разработки алгоритма и программы для автоматизации определения рациональных режимов обработки на плоскошлифовальных станках появилась, потому что традиционный поиск справочно-нормативных данных замедляет технологическую подготовку производства, повышает вероятность случайных ошибок при расчете величин, использующих таблицы, заставляет выполнять много рутинной и однообразной работы.

Задача автоматизации определения оптимальных режимов обработки на плоскошлифовальных станках относится к трудно формализуемой задаче, требующей большого объема исследований. Наиболее удобным для автоматизации вариантом существующего методического обеспечения этой задачи является методика, изложенная в [1].

Проанализировав информационные структуры, необходимые для автоматизации режимов резания при плоском шлифовании были разработаны информационно-логические модели этих структур и выбраны средства их реализации.

Расчет режимов резания непосредственно должен быть связан с конкретным станком, поэтому предусмотрена возможность создания и пополнения базы данных о характеристиках моделей станков выбранной группы.

Проанализировав различные подходы к проектированию систем, для автоматизация определения рациональных режимов обработки на плоскошлифовальных станках был выбран метод объектно-ориентированного проектирования, а инструментальным средством выбрана система программирования Turbo Delphi.

В программе реализовано определение рациональных режимов обработки на плоскошлифовальных станках для следующих типов обработки: плоское шлифование торцом круга на станках с круглым столом; плоское шлифование торцом круга на вертикальных двухшпиндельных станках с круглым столом; плоское шлифование периферией круга на станках с круглым столом.

Ввод исходных данных осуществляется при помощи выбора нужного значения из раскрывающегося списка или выбора нужной кнопки, при этом непосредственный ввод сведен к минимуму, что избавляет пользователя от случайных ошибок.

На рисунке 1 представлено окно плоского шлифование торцом круга на станках с круглым столом.

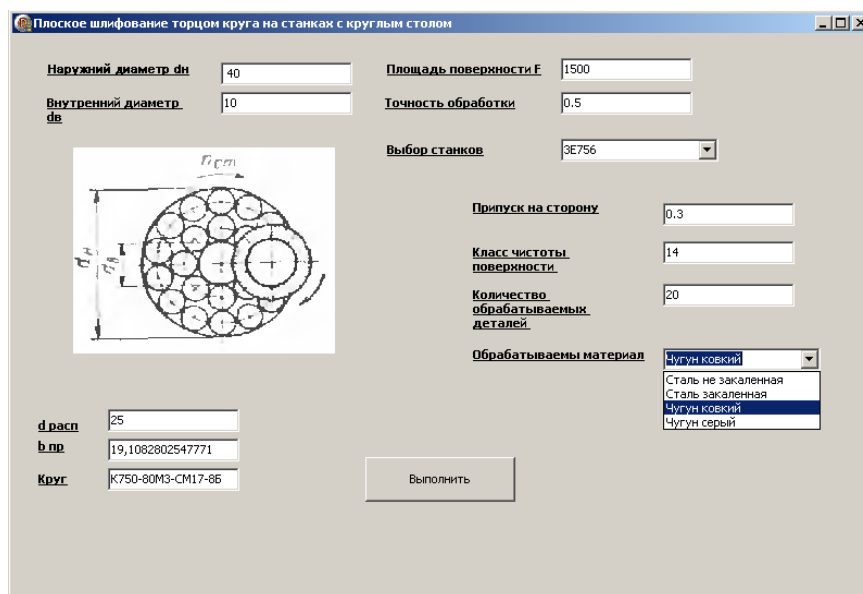


Рисунок 1 – Плоское шлифование торцом круга на станках с круглым столом

Основным результатом разработанной программы является инструкционная карта, в которой указано наименование операции, условия обработки, сведения о станке и характеристика шлифовального круга, оптимальные режимы резания, а также основное технологическое (машинное) время, необходимое для выполнения операции.

## Литература

1. Режимы резания металлов. Справочник. Под ред. Ю.В. Барановского. – М.: Машиностроение, 1972. – 408 с.

**Я.А. Юницкий (УО «ГГУ им. Ф.Скорины», Гомель)**

Науч. рук. **А.В. Воружев**, канд. техн. наук, доцент

## ПРИМЕНЕНИЕ ТЕХНОЛОГИИ NVIDIA CUDA ДЛЯ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ

Устройства для превращения персональных компьютеров в миниатюрные суперкомпьютеры известны довольно давно. В 80-х гг. прошлого века на рынке они предлагались под названием «транспьютеры». Но в последнее время эстафета параллельных вычислений перешла к массовому рынку, так или иначе связанному с трёхмерными играми.

Графическая составляющая современных приложений неуклонно стремиться к совершенству – фотореализму. Но для столь сложного рендера в реальном времени необходимо максимальное быстродействие вычислительного устройства. Графические адаптеры с многоядерными процессорами для параллельных векторных вычислений, используемых в 3D-графике, достигают высокой пиковой производительности, которая универсальным процессорам не под силу.

В конечном итоге случилось то, что должно было – графические адаптеры стали применяться для неграфических расчетов. Современные видеочипы содержат сотни математических исполнительных блоков, и вся эта мощь может быть использована для ускорения различных «сильных» приложений.

Компания NVIDIA в 2007 г. представила технологию CUDA. Технология CUDA – это программно-аппаратная вычислительная архитектура NVIDIA, основанная на расширении языка Си, которая даёт возможность организации доступа к набору инструкций графического ускорителя и управления его памятью при организации параллельных вычислений. В основе API лежит расширенный язык Си, а для трансляции кода с этого языка в состав CUDA SDK входит собственный компилятор.

Основные характеристики CUDA:

- унифицированное программно-аппаратное решение для параллельных вычислений на видеочипах NVIDIA;
- большой набор поддерживаемых решений;
- стандартный язык программирования Си;
- стандартные библиотеки численного анализа FFT (быстрое преобразование Фурье) и BLAS (линейная алгебра);
- оптимизированный обмен данными между CPU и GPU;
- взаимодействие с графическими API OpenGL и DirectX;
- возможность разработки на низком уровне.

Преимущества CUDA:

- интерфейс программирования приложений CUDA основан на стандартном языке программирования Си с расширениями;
- CUDA обеспечивает доступ к разделяемой между потоками памяти размером в 16 Кб на мультипроцессор, которая может быть использована для организации кэша с широкой полосой пропускания;
- более эффективная передача данных между системной и видеопамятью;
- отсутствие необходимости в графических API с избыточностью и накладными расходами;
- линейная адресация памяти, возможность записи по произвольным адресам;
- аппаратная поддержка целочисленных и битовых операций.

Основные ограничения CUDA:

- отсутствие поддержки рекурсии для выполняемых функций;
- минимальная ширина блока в 32 потока;
- закрытая архитектура CUDA, принадлежащая NVIDIA.

Основные преимущества CUDA вытекают из того, что эта архитектура спроектирована для эффективного использования неграфических вычислений на GPU и использует язык программирования C, не требуя переноса алгоритмов в удобный для концепции графического конвейера вид. CUDA предлагает способ вычислений на GPU без применения графических API, предлагающий произвольный доступ к памяти (scatter или gather). Такая архитектура использует все исполнительные блоки, а также расширяет возможности за счёт целочисленной математики и операций битового сдвига.

В среднем, при переносе вычислений на GPU, во многих задачах достигается ускорение в 5-30 раз, по сравнению с быстрыми универсальными процессорами. Самые большие цифры достигаются на коде, который не очень хорошо подходит для расчётов при помощи блоков SSE, но вполне удобен для GPU.

Некоторые примеры ускорений синтетического кода на GPU против SSE кода на CPU (по данным NVIDIA):

- флуоресцентная микроскопия: 12x;
- молекулярная динамика (non-bonded force calc): 8–16x;
- электростатика (прямое и многоуровневое суммирование Кулона): 40–120x и 7x.

Для пользователей, не имеющих отношения к программированию, CUDA так же может быть полезна, например, для ускорения рендеринга видео или 3D графики, а также для увеличения частоты обновления изображения в современных компьютерных играх. Все современные программы нелинейного видеомонтажа имеют поддержку CUDA.

**Я.А. Юницкий (УО «ГГУ им. Ф.Скорины», Гомель)**

Науч. рук. **А.В. Воружев**, канд. техн. наук, доцент

## **НАПИСАНИЕ ПРИЛОЖЕНИЯ ИДЕНТИФИКАЦИИ ПРОЦЕССОРА С ПОМОЩЬЮ ИНСТРУКЦИИ CPUID**

CPUID – ассемблерная мнемоника инструкции процессоров x86, используется для получения информации о процессоре. Используя её, программа может определить тип процессора и его возможности. Процессоры поддерживают инструкцию CPUID (как Intel, так и AMD), начиная с пятого поколения (Pentium) и поздних моделей i486.

Данная инструкция используется в любом диагностическом ПО, а также в приложениях, поддерживающих различные наборы процессорных инструкций.

Инструкция работает с регистрами данных общего назначения EAX, EBX, ECX и EDX. Входное значение записывается в регистр EAX, и после вызова команды CPUID во все четыре регистра записываются выходные данные, доступные для обработки.

Условимся, что разрабатываемое приложение должно определять наименование процессора (производитель, модель, частота), строковый идентификатор производителя, а также выводить все значения регистров в ранее определенных пределах. Так как все современные, или сравнительно современные, процессоры поддерживают команду CPUID, производить проверку поддержки не имеет смысла.

Команда вызывается на языке Ассемблера, либо при помощи ассемблерных вставок. Функция, вызывающая CPUID, в среде Embarcadero Rad Studio для языка C++, может выглядеть следующим образом:

```
void cpuid(unsigned int *a, unsigned int *b, unsigned int *c,
unsigned int *d, unsigned int InfoType)
{
    unsigned int a1, b1, c1, d1;
    asm xor EBX, EBX           //Очистка регистров
    asm xor ECX, ECX
    asm xor EDX, EDX
    asm mov EAX, InfoType //Передача входного значения
    asm cpuid                 //Вызов команды CPUID
    asm mov a1, EAX           //Передача возвращаемых значений
    asm mov b1, EBX
    asm mov c1, ECX
    asm mov d1, EDX
    *a = a1; *b = b1; *c = c1; *d = d1;
}
```

Для определения идентификатора производителя следует передать регистру EAX значение 0, при этом сам идентификатор вернётся в виде кодов ASCII символов соответственно в регистрах EBX, EDX и ECX, начиная с младшего байта каждого регистра. Для процессоров Intel таким идентификатором является строка «GenuineIntel». Регистр EAX в таком случае вернёт максимальное входное значение доступное для данного процессора в рамках базового набора функций.

Строковое наименование процессора возвращают все четыре регистра, при использовании расширенного набора функций. Для его получения, следует установить 31 бит в положение 1 и передать это

значение регистру EAX. Он в свою очередь вернёт максимальное доступное значение входного параметра расширенного набора функций. Само значение можно получить последовательно передав регистру EAX значения 80000002h – 80000004h. Выходные данные также представляют собой последовательность ASCII символов хранящиеся в регистрах от EAX до EDX.

Пример результата работы такого приложения представлен на рисунке 1:

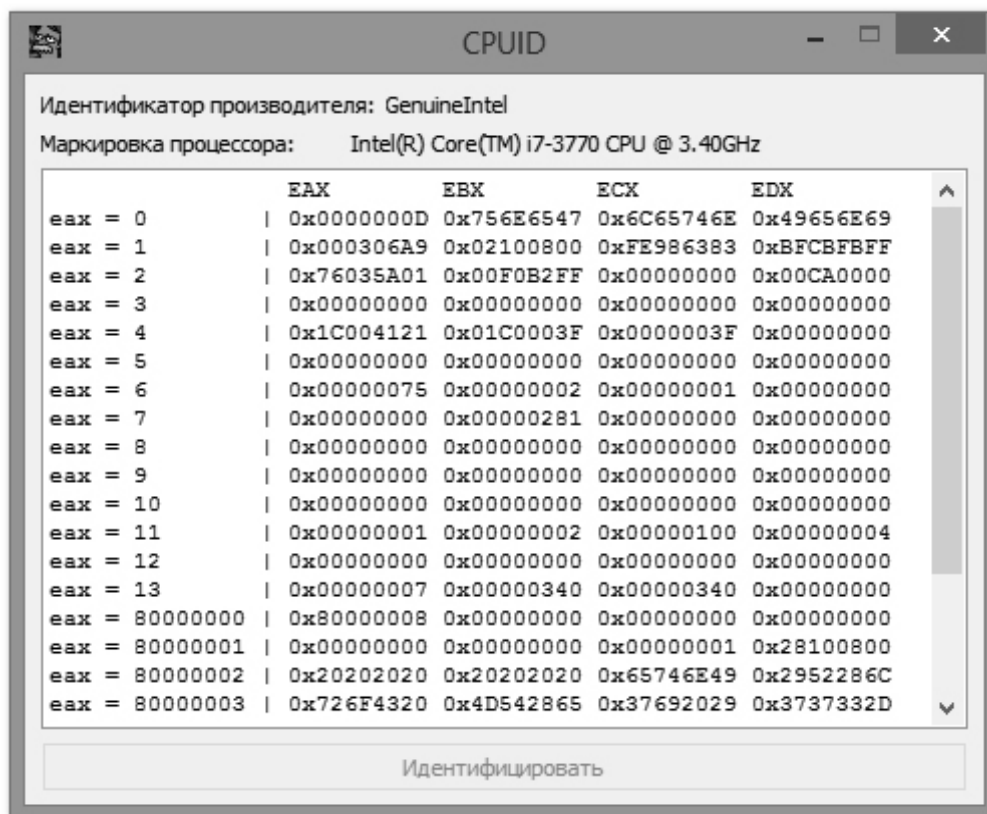


Рисунок 1 – Пример приложения использующее инструкцию CPUID

**А.С. Якубова (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **М.С. Данильченко**, ассистент

**СФЕРИЧЕСКИЕ ПАНОРАМЫ МУЗЕЯ  
 «ЛЁС САЛДАТА» г. РОГАЧЕВА**

Сферическая панорама экспозиции позволяет составить наиболее полное впечатление о любом музее. В качестве объекта был выбран музей города Рогачев – «Лёс салдата».

Музей «Лёс салдата» создан по инициативе советского поисковика Горевго Марата Григорьевича при Рогачевском городском центре



поисковой и краеведческой работы и был открыт решением Рогачевского горисполкома № 79 от 29 февраля 1994 г. В основу экспозиции положен поисковый материал отряда «Разведчики военной славы», действующего на базе ПТУ № 56 им. П.Я. Головачева города Гомель.

Процесс создания сферических панорам состоит из трех последовательных этапов:

- съемка – производится панорамная съемка объектов, по которым будет сделана панорама;
- обработка фотоматериала: коррекция цветов, ретушь, удаление дефектов и нежелательных элементов на фото;
- склейка фотопанорам.

На первом этапе была произведена съемка помещения. Фокусное расстояние было выставлено самое малое, что в данном случае равно 18mm. Фокусировка проведена так, чтобы у всех объектов съемки была необходимая резкость. В ручном режиме также настраивалась и экспозиция. Чтобы глубина резкости фотографий была достаточной, число диафрагмы установлено на более высокое значение – F9.0 (рисунок 1).



Рисунок 1 – Настройки фотоаппарата в ручном режиме

Выдержка выставлена так, чтобы на снимке не было засвеченных и слишком темных областей. Исходя из освещения кабинета, значение ISO установлено на отметке 3200 и зафиксирован баланс белого.

Далее последовательно снимался кадр за кадром, степень перекрытия соседних кадров составляла приблизительно 30 %. После того, как были получены изображения, они были загружены в программу для создания панорам – AutoPano. При необходимости положение автоматически выставленных контрольных точек несложно скорректировать, а также добавить новые контрольные точки либо удалить ненужные (рисунок 2).

Здесь склейка происходит в полностью автоматическом режиме с использованием собственных алгоритмов SIFT и RANSAC, также

в программе есть полная поддержка HDR. Дисторсия оптики корректируется автоматически, а смешивание снимков производится средствами предварительно указанного в настройках модуля, в перечне которых имеется и плагин Smartblend.

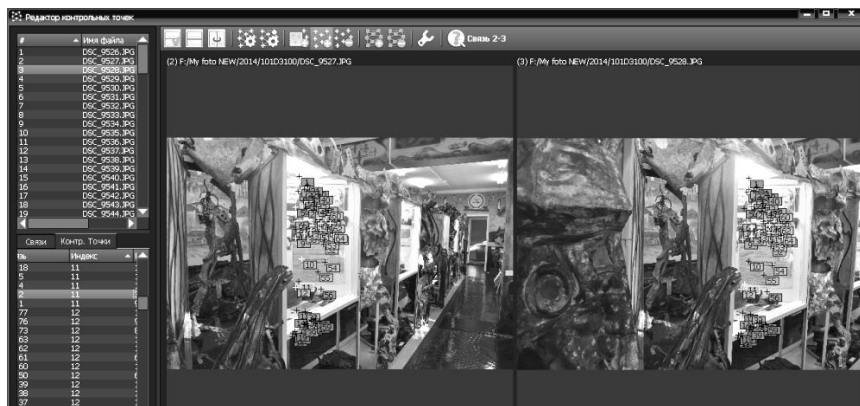


Рисунок 2 – Добавление контрольных точек в Autorano

После редактирования панорамы ее можно сохранить как изображение. Для этого нужно нажать выполнить сборку изображения. После нажатия кнопки сборка появится пакетный сборщик.

После сборки панораму можно найти в указанной папке. Готовые панорамы приведены на рисунках 3–4.



Рисунок 3 – Панорама музея «Лёс салдата», вид 1



Рисунок 4 – Панорама музея «Лёс салдата», вид 2

**А.С. Якубова (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **М.С. Данильченко**, ассистент

## **ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ СБОРКИ ПАНОРАМНЫХ ИЗОБРАЖЕНИЙ**

Панорамная фотография позволяет составить наиболее полное впечатление об окружающей среде. Фотопанорамы обычно создаются из нескольких специально подготовленных перекрывающихся фотографий особыми программами, которые «сшивают» снимки.

Сшивание снимков в виртуальные панорамы производится в специальных программах-сшивателях, как правило, умеющих сшивать в автоматическом, полуавтоматическом либо ручном режиме. Технология сшивания во всех программах примерно одинакова. Вначале нужно загрузить исходный набор изображений и установить порядок их размещения, произвести коррекцию изображений. Потом снимки преобразуются к определенной проекции, например сферической, – это необходимо, поскольку только в таком виде снимки можно сшивать. Затем осуществляется состыковка полученных изображений между собой – путем установки контрольных точек для каждой из пар граничащих друг с другом снимков. На последнем этапе смежные изображения смешиваются с целью выравнивания их яркости, контрастности и цветовой тональности. Результатом всех этих действий станет получение фотопанорамы. Для превращения ее в виртуальную потребуется конвертировать панораму в соответствующий формат, а для демонстрации виртуальной панорамы в Интернете ее нужно будет вручную встроить в файл существующей интернет-страницы либо сгенерировать шаблонную веб-страницу.

Программа PTGui является одним из лидеров среди программ, предназначенных для создания панорам. Изначально программа разрабатывалась как интерфейс для популярного инструмента создания панорам Panorama Tools. Начиная с версии 5.0, программа может использоваться двояко – как интерфейс для Panorama Tools и как отдельное приложение для сшивания картинок.

В PTGui предусмотрено два режима работы – простой и продвинутый. Между режимами можно переключиться в любой момент. Простой режим позволяет создавать панорамы автоматически и требует минимум ручной работы. Продвинутый дает полный контроль над процессом создания панорамы, позволяя изменять параметры.

PTGui умеет автоматически расставлять контрольные точки, спрямлять горизонт, корректировать цвет и экспозицию. На рисунке 1 представлено окно программы.

При работе в простом режиме панорама создается за три шага: загрузить файлы с картинками, выполнить анализ картинок, создать панораму. Первый и третий: нажать кнопку – выбрать файлы, нажать кнопку – создать результирующий файл.

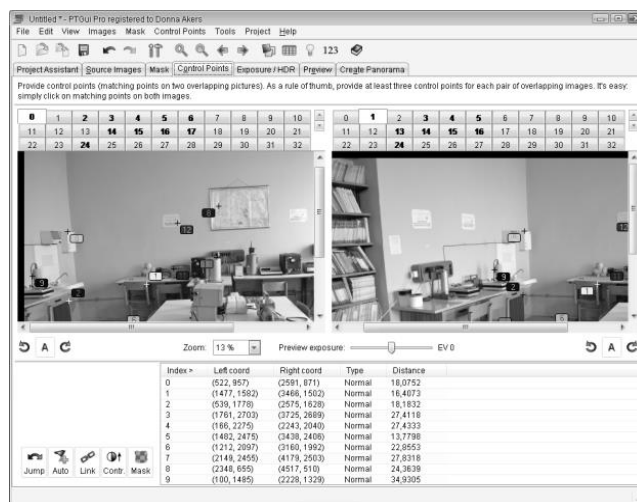


Рисунок 1– Окно программы PTGui

Второй шаг – процесс анализа – также автоматизирован: программа самостоятельно находит и расставляет контрольные точки, определяет расположение фотографий друг относительно друга, выставляет тип панорамы, выбирает оптимальный режим спрямления горизонта. Можно редактировать контрольные точки, корректировать параметры экспозиции и цвета. В продвинутом режиме доступных для редактирования параметров гораздо больше. Изменять можно все. Появляется возможность задать параметры линз фотоаппарата, обрезать края изображения, просмотреть детальную информацию о каждой картинке, явно указать, какие алгоритмы использовать для слияния, интерполяции, смешивания изображений и т. д.

Готовая панорама – это результирующая картинка, собранная из исходных картинок. PTGui позволяет сохранить панораму в нескольких разных форматах. Во-первых, в виде единой картинки в формате JPEG или TIFF. Во-вторых, в виде PSD-файла Adobe Photoshop в многослойном виде – каждая исходная картинка будет размещаться в результирующем файле на отдельном слое. В-третьих, панораму можно сохранить в виде нескольких картинок. Наконец, панораму можно конвертировать в формат QuickTime VR (QTVR), специально предназначенный для панорам.

Ещё одним продуктом для создания фотопанорам является Autorano. Обеспечивает создание цилиндрических, плоских и сферических панорам из неограниченного числа снимков.

Склейка панорам в Autorano производится в полностью автоматическом режиме: программа самостоятельно извлекает из указанной папки фотографии, которые являются частью панорамы, игнорируя ненужные снимки, определяет параметры объектива и сшивает изображения, расставив контрольные точки. При этом у исходных изображений может быть произведена обрезка и коррекция цвета. Дисторсия оптики корректируется автоматически, а смешивание снимков производится средствами предварительно указанного в настройках модуля, в перечне которых имеется и плагин Smartblend. При необходимости положение автоматически выставленных контрольных точек несложно скорректировать.

Собранную панораму можно повернуть, обрезать, изменить у нее проекцию и положение центра, откорректировать горизонт, цвет (возможна HDR-цветокоррекция) и пр. На рисунке 2 показан интерфейс программы.

Готовые панорамные снимки сохраняются в форматах PSD, JPG, PNG, TIFF, HDR или EXR, возможен экспорт в формат Flash. Реализован экспорт проектов Autorano в PanoTools, а также импорт из PanoTools в Autorano, что позволяет использовать возможности этих двух решений одновременно: первое, например, полезно для получения более точной оптимизации в сложных случаях.

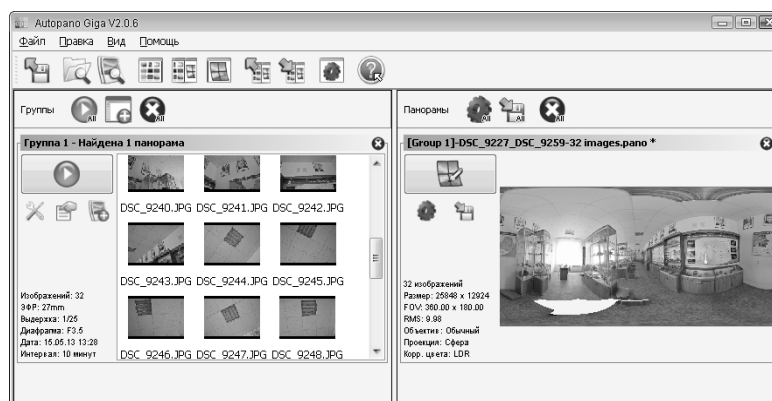


Рисунок 2 – Autorano Giga

Pano2VR – удобный инструмент для конвертирования готовых панорам в формат Quicktime VR (QTVR) или Macromedia Flash 8/9/10 с одновременным формированием связанных с ними HTML-файлов (рисунок 3). В качестве исходных панорам могут быть использованы панорамы в форматах JPEG, PNG или TIFF (поддерживаются проекции Cylindrical, Equirectangular, Cube Faces и др.) либо виртуальные QTVR-панорамы. В программе имеется возможность трансформации панорамных изображений в сферу, зеркальный шар,

цилиндр, горизонтальный крест и т. п. Возможно добавление в виртуальную панораму активных зон для перехода на конкретные веб-страницы, например, с другими панорамами, что позволяет получать упрощенные виртуальные туры.

**С.С. Якубова (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **М.С. Данильченко**, ассистент

## **ПОЛУЧЕНИЕ ИЗОБРАЖЕНИЙ ДЛЯ ФОТОПАНОРАМ**

Фотопанорама или панорамная фотография – это фотография, имеющая большой угол обзора, то есть демонстрирующая вид местности из одной точки (в которой находилась камера) во многих направлениях. Фотопанорамы обычно создаются из нескольких специально подготовленных перекрывающихся фотографий особыми программами, которые «сшивают» снимки в единую панораму.

Панорамная съемка может быть выполнена двумя способами: круговым (секторным) и линейным. Круговое панорамирование применяется в тех случаях, когда фотографируемые объекты расположены в разных плоскостях или под углом друг к другу, а также когда их удобнее зафиксировать из одной точки. При этом аппарат укрепляют на специальной панорамной или легко поворачиваемой вокруг вертикальной оси универсальной головке штатива.

В случаях, когда объекты находятся на одной линии (фасады домов, дороги, стены, заборы и т. п.), более предпочтительна фото-съемка методом линейной панорамы с перемещением аппарата по прямой параллельной линии.

Для съемки панорам в фотоаппарате должна быть предусмотрена функция фиксации экспозиции – ручной режим установки выдержки и диафрагмы, а так же ручной режим установки баланса белого и дистанции фокусировки. Объектив должен иметь фокусное расстояние от 24 до 80 мм, более короткофокусные объективы дают большие геометрические искажения.

При съёмке центр кадра в видоискателе должен всегда совпадать с горизонтом. Одним из важных моментов создания панорамы является необходимость вращения камеры вокруг нодальной точки объектива. Нодальная точка – это точка внутри объектива камеры, где пересекаются лучи света, идущие к матрице. При вращении камеры вокруг этой точки отсутствует параллакс объектов. Параллакс – смещение объектов переднего плана относительно объектов заднего плана при

повороте камеры. Влияние параллакса на снимок показано на рисунке 1. Видно, что при повороте камеры происходит смещение объектов ближнего и дальнего плана относительно друг друга. Такое смещение может вызвать трудности при сшивании панорамы. Чтобы избежать подобных проблем используется штатив с панорамной головкой.

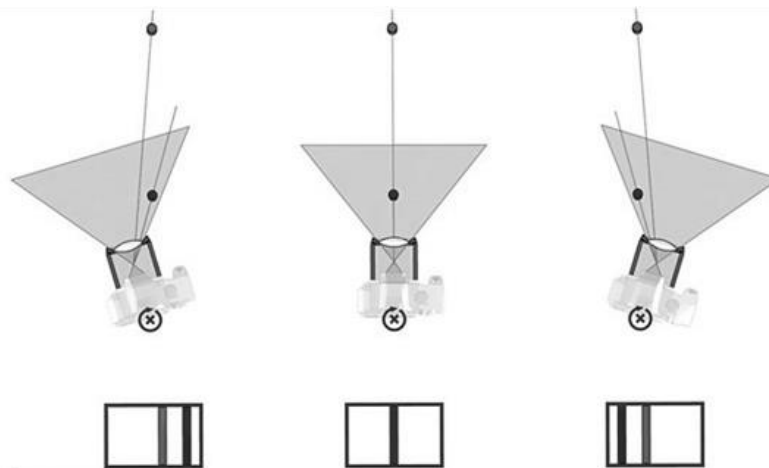


Рисунок 1 – Пример параллакса

Снимать каждый последующий кадр нужно так, чтобы он перекрывал предыдущий примерно на 30% или больше, больший процент перекрытия означает лучшее качество сборки готовой фотопанорамы (рисунок 2). Необходимо следить, чтобы линия горизонта оставалась неизменной, контролировать это позволяют пузырьковые уровни на штативе. Кроме того, места швов лучше располагать на достаточно однотонных объектах.



Рисунок 2 – Пример перекрытия кадров

При съемке важно чтобы освещение не менялось. Движущиеся объекты желательно полностью помещать в границах одного кадра, иначе может возникнуть ситуация, когда на одном фрагменте половина объекта есть, а на втором нет.

Снимать следует с максимально возможной глубиной резкости (наведение резкости нужно осуществлять вручную и так, чтобы глубина

резкости была достаточной для проработки всех визуально-активных объектов, f8 и меньше будет в самый раз.) и в едином режиме – то есть с одинаковой экспозицией и одинаковым балансом. Рекомендуется определить нормальную экспозицию самого яркого и самого тёмного фрагментов, и при съёмке использовать осреднённую. Так, например, если для самого яркого фрагмента панорамы экспозара будет 250/f8.0, а для тёмного – 60/f8.0, то получим искомую экспозару 125/f8.0.

Лучше снимать без светофильтров. Потенциальными проблемами может быть обесцвечивание или затемнение изображения по краям кадра. Кроме этого в кадре могут появляться блики, особенно это заметно при съёмке на ярком солнце. Отдельно стоит упомянуть поляризационный фильтр, он, безусловно, может улучшить качество фотографии, но при съёмки панорам, при изменении угла объектива к солнцу могут происходить искажения цветов.

Использование автоматической фокусировки при съёмке панорамы может привести к тому, что отдельные кадры сфокусируются по-разному, или, еще хуже, некоторые кадры окажутся совсем вне фокуса, тогда собрать панораму будет невозможно. При работе в простом режиме панорама создается за три шага: загрузить файлы с картинками, выполнить анализ картинок, создать панораму. Первый и третий: нажать кнопку – выбрать файлы, нажать кнопку – создать результирующий файл.

**С.С. Якубова (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **М.С. Данильченко**, ассистент

## **ВИРТУАЛЬНЫЙ ТУР ПО ЗИМНЕМУ САДУ**

Сферические 3D панорамы и виртуальные туры – это удобная технология, позволяющая передать полную картину окружающего пространства. На основе панорамных фото собираются связанные между собой переходы – виртуальные туры. При помощи звуковых и других визуальных эффектов можно передать не только сведения об объекте, но и его атмосферу.

Процесс создания виртуального тура состоит из трех последовательных этапов: съёмка – производится панорамная съёмка объектов, по которым будет сделан виртуальный тур; обработка фотоматериала: коррекция цветов, ретушь, удаление дефектов и нежелательных элементов на фото; создание виртуального тура – склейка фотопанорам, монтаж виртуального тура.



Для съёмки использовался цифровой зеркальный фотоаппарат Nikon D3100 с объективом AF-S DX NIKKOR 18–55 мм VR со стабилизатором. Штатив использовался со встроенным уровнем, поскольку ключевым моментом при съёмке панорамы является строгое вертикально-горизонтальное положение всех осей (рисунок 1).



Рисунок 1 – Штатив с уровнем

Фотоаппарат был выставлен строго горизонтально. Далее последовательно снимался кадр за кадром, степень перекрытия соседних кадров составляла 30 %. Последними были сделаны кадры зенита и надира. Для этого фотоаппарат поставили на штативе вертикально вниз, затем отсоединили, убрали штатив и, удерживая камеру на вытянутой руке в таком же положении, сделали снимок надира. Аналогично был сделан кадр зенита. После того, как были получены изображения, можно переходить к склеиванию их в панораму. Работу начинали с загрузки в PTGui фотографий, предназначенных для склейки. Далее генерируем контрольные точки и выполняем оптимизацию. В результате получаем эквидистантную проекцию сферической панорамы, которая представлена на рисунке 2.



Рисунок 2 – Сферическая панорама

Отличительной особенностью виртуального тура является возможность перемещения от панорамы к панораме. Она реализуется

с помощью, так называемых, точек перехода, которые позволяют «путешествовать» по виртуальному туру в зависимости от желания зрителя.

Для реализации виртуального тура была использована программа Panotour Pro. Были добавлены сразу несколько панорам. Для каждой панорамы свой набор настроек. Для того чтобы создать переходы, установлены точки перехода (рисунок 3). Чтобы сделать виртуальный тур более удобным и информативным добавлены элементы:

- горячая точка (hotspot);
- кнопка и панель управления;
- интерактивный список панорам и миниатюры;
- карта виртуального тура и радар;
- всплывающие окна с текстом, видео;
- звуковое сопровождение.

Горячая точка это изображение, текст или область панорамы, которые представляет собой ссылку. При наведении курсора на горячую точку всплывает подсказка, а нажатие активирует загрузку соответствующей 3D панорамы или открывает всплывающее окно. Горячих точек на одной панораме может быть несколько, они логически соответствуют направлению движения, которое мы используем в реальности.

Карта виртуального тура позволяет облегчить ориентацию в пространстве цифровой реальности. А горячие точки, размещённые на карте, имеют дополнительную функцию – радар. Радаром называют цветную зону, вращающуюся вокруг горячей точки. Она указывает направление, которое в данный момент соответствует изображению на 3D панораме.



Рисунок 3 – Скриншот виртуального тура

Всплывающее окно является важным элементом, который позволяет сделать виртуальный тур более информативным. А размещение информации проходит без ущерба для просмотра самой панорамы,

так как после загрузки всплывающие окна скрыты от пользователя и появляются только после того, как мы его активируем. Открытие и закрытие всплывающих окон может сопровождаться различными спецэффектами.

**Д.А. Ярошенко, (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **А.И. Кучеров**, ст. преподаватель, магистрант техн. наук

## **РАЗРАБОТКА ИНСТРУМЕНТАРИЯ ПО УПРАВЛЕНИЮ СУБД DB2 В СРЕДЕ ОС LINUX**

Поскольку система баз данных DB2 может быть весьма большой и может иметь много пользователей, должно существовать лицо или группа лиц, управляющих этой системой. Такое лицо называется администратором базы данных.

В любой базе данных должен быть хотя бы один человек, выполняющий административные обязанности. Если база данных большая, эти обязанности могут быть распределены между несколькими администраторами.

В обязанности администратора могут входить:

- инсталляция и обновление версий сервера DB2 и прикладных инструментов;
- распределение дисковой памяти и планирование будущих требований системы к памяти;
- создание первичных структур памяти в базе данных (табличных пространств) по мере проектирования приложений разработчиками приложений;
- создание первичных объектов (таблиц, представлений, индексов) по мере проектирования приложений разработчиками;
- модификация структуры базы данных в соответствии с потребностями приложений;
- добавление пользователей и поддержание защиты системы;
- управление и отслеживание доступа пользователей к базе данных;
- отслеживание и оптимизация производительности базы данных;
- импорт и экспорт данных из других БД или носителей;
- планирование резервного копирования и восстановления;
- поддержание архивных данных на устройствах хранения информации;
- осуществление резервного копирования и восстановления.

В некоторых случаях база данных отдельно может иметь одного или нескольких сотрудников службы безопасности, отвечающих за регистрацию новых пользователей, управление и отслеживание доступа пользователей к базе данных, и защиту базы данных.

Многие из действий выполняемых этими людьми довольно рутинны, и для упрощения и ускорения выполнения таких операций в Unix-подобных ОС используются специальные Shell-скрипты (для Linux это Bash-shell, для систем на базе IBM AIX – Korn-shell).

BASH – Bourne-Again SHell (что может переводиться как «перерожденный шел»), самый популярный командный интерпретатор в Unix-подобных системах, в особенности в GNU/Linux. Но BASH не только командная оболочка, это еще и превосходный скриптовый язык программирования, который позволяет, к примеру, быстро написать простенький скрипт для выполнения ежедневной (-недельной, -месячной) рутинной работы или, скажем, для бэкапа директории.

Небольшой пример скрипта, выполняющего генерацию пароля из N букв с чтением ввода с клавиатуры:

```
#!/bin/bash
# генерации пароля из N букв
read N
    PASSWORD=`cat /dev/urandom | tr -d -c 'a-zA-Z0-9'
| fold -w $N | head -1`
echo Password=$PASSWORD
```

Администратор может создать под себя отдельный скриптовый \*.sh файл, который будет включать часто используемые команды и действия, что в свою очередь позволит сократить количество ввода текстовых команд, уменьшит вероятность ошибки и сохранит его время.

Также существуют различные специализированные инструменты для облегчения работы администраторов. К ним можно отнести IBM UrbanCode Deploy. Это специализированное ПО для организации и автоматизации развертывания приложений, настройки промежуточного ПО и изменения баз данных в различных средах, их внедрения и эксплуатации. Это ПО позволяет развертывать приложения с нужной частотой по требованию или по расписанию, а также используя самообслуживание, что существенно упрощает работу администраторов.

Возможности UrbanCode Deploy:

Автоматизация взаимодействия и развертывания — создание процессов автоматического развертывания с использованием графического редактора, запуск процессов на тысячах машин одновременно.

Управление средой и настройками – использование мощных функций управления, позволяющих использовать разные типы развертывания: только с настройками или с программным кодом и настройками.

Хранилище артефактов — более безопасное хранение артефактов, полученных при развертывании, позволяющее отслеживать процессы.

Управление настройками промежуточного ПО – определение настроек IBM WebSphere Application Server, DataStage и DB2.

Безопасность, подтверждения и уведомления – настройка разрешений для пользователей, определение входных критериев среды и информирование заинтересованных лиц о действиях в процессе развертывания.

Работа с IBM UrbanCode Deploy осуществляется посредством Web-интерфейса в окне браузера, благодаря чему в случае необходимости администратор может получить доступ к приложению из любой точки земного шара. В случае если какого-то функционала не хватает – всегда можно написать свой shell-скрипт, который посредством UrbanCode можно запустить на стороне сервера.

**Т.Н. Яцкова (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч.рук. **В.Н. Кулинченко**, ст. преподаватель

## **РАЗРАБОТКА ПРОЕКТА МОДЕРНИЗАЦИИ ИНФОРМАЦИОННОЙ СИСТЕМЫ УЧЕТА КЛИЕНТОВ ТЕРРИТОРИАЛЬНОГО КАЗНАЧЕЙСТВА ПО ГОМЕЛЬСКОЙ ОБЛАСТИ**

Одной из задач, которая всегда стоит перед любой действующей организацией – это задача по повышению эффективности работы, нахождение более оптимальных путей выполнения своих функций. Так и для территориальных органов казначейства можно рассматривать повышение эффективности в свете процесса информатизации, как одно из направлений развития Республики Беларусь. В процессе информатизации присутствуют, как социально-экономический, так и научно-технический аспекты, организованные на базе информационных систем и сетей с последующим формированием и накоплением информационного ресурса, а также выделением потребителей ресурса и делегирования им прав на его использование.

Информационный ресурс (ИР) – интегрированный комплекс данных, информации и моделей, организованный и используемый для решения проблемных ситуаций в конкретных областях деятельности,

заменяющих материальные (овеществленные), трудовые, финансовые, энергетические, временные и другие виды ресурсов.

Казначейская система исполнения бюджета в Республике Беларусь была изначально создана с применением передовых информационных технологий и мирового опыта по созданию систем в сфере управления государственными финансами. Поэтому одним из стержневых моментов ее деятельности является интеграция финансовых и компьютерных технологий в единое целое.

Один из первых этапов заключается во внедрении электронного документооборота между территориально распределенными госорганами и организациями с применением средств криптографической защиты информации и электронной цифровой подписи. С этой целью по инициативе Главного государственного казначейства республиканским унитарным предприятием «Информационно-вычислительный центр Министерства финансов Республики Беларусь» создана программа «Клиент территориального казначейства» (Клиент ТК), которая предназначена для использования как территориальными органами государственного казначейства, так и получателями бюджетных средств. Задача «Клиент ТК» позволяет выполнять следующие функции:

- загрузку платежных документов, подготовленных в бухгалтерском комплексе в виде файлов согласованной структуры;
- ввод, контроль и отправку в территориальный орган государственного казначейства платежных документов и запросов на получение справочной, оперативной, отчетной информации;
- прием, отображение, выгрузка в Excel и печать справочной, оперативной и отчетной информации от территориального органа государственного казначейства;
- применение средств криптографической защиты информации;
- учет и хранение электронных документов;
- автоматическую рассылку обновлений программного обеспечения для пользователей.

Внедренная система облегчила аналитическую составляющую работы. В функционал «Клиент ТК» включен инструмент оперативного получения информации об объеме уточненных плановых назначений в разрезе статей экономической классификации, суммах заключенных договоров с учетом еще неиспользованного остатка денежных средств. Оперативное предоставление этих данных позволяет проводить детальный анализ исполнения бюджетной сметы на любую дату в разрезе необходимых элементов бюджетной классификации (параграфов, статей, подстатей, элементов), осуществлять мониторинг полноты освоения выделенных бюджетных ассигнований, помогает

отслеживать соблюдение договорных обязательств сторонними организациями на основании представляемой задачей динамики проведенных платежей.

Результат усилий по разработке программного обеспечения состоит в передаче в эксплуатацию программного продукта, удовлетворяющего требованиям пользователей. Соответственно, в процессе эксплуатации продукт будет изменяться или эволюционировать. Сопровождение программного продукта заключается в первую очередь в удовлетворении потребностей пользователя, выявление и устранение обнаруженных дефектов и ошибок, введение новых функций и компонентов в программный продукт, анализ состояния и корректировка документации.

Для обеспечения учета и отслеживания возникающих в процессе эксплуатации вопросов и замечаний со стороны бюджетных организаций работающих с задачей «Клиент территориального казначейства» разработана система по её сопровождению.

Разработанная автоматизированная система позволяет:

- оперативно находить информацию об организации (УНК, УНП, полное наименование, контактные данные);
- оперативно находить решения используя справочник типовых вопросов;
- ввести учет договоров;
- контролировать поступление и последующую передачу актов выполненных работ согласно заключенного договора на сопровождение ПО;
- реализована возможность печати договора с заполненными реквизитами по организации;
- структурировать и визуализировать статистическую информацию на основании справочника учета звонков;
- получать информацию по клиентам территориального казначейства: отчеты по версиям используемого ПО, перечень поступивших вопросов как по клиенту в частности, так и в целом по все организациям; сведения о способах обменом информацией (коммутируемый доступ по телефонным номерам или VPN РУП «Белтелеком»).

Общая структура программы представлена на рисунке 1.

Разработанное автоматизированное место по учету и сопровождению клиентов территориального казначейства позволяет оперативно отвечать на поступающие вопросы от организаций и эффективно решать поставленные задачи по модернизации программного комплекса, тем самым удовлетворив все потребности пользователя, что является одной из основных задач инженера по сопровождению любого программного комплекса.

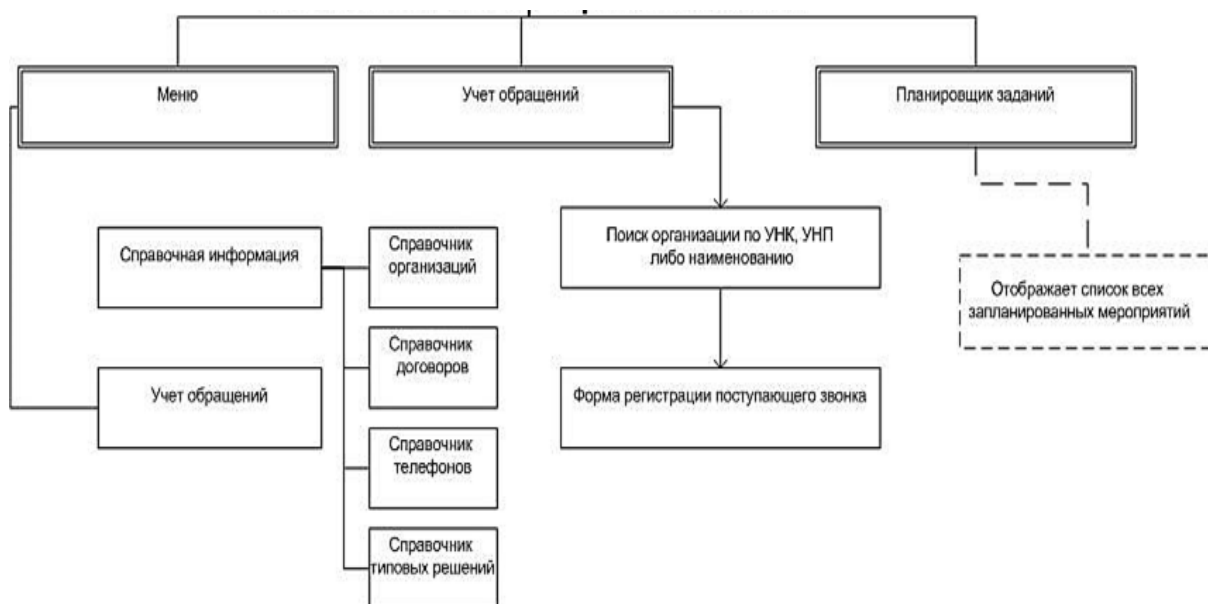


Рисунок 1 – Главное окно приложения

**Т.Н. Яцкова (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **В.Н. Кулинченко**, ст. преподаватель

## **АВТОМАТИЗАЦИЯ РАБОЧЕГО МЕСТА ПО СОПРОВОЖДЕНИЮ ПРОГРАММНОГО КОМПЛЕКСА УЧЕТА КЛИЕНТОВ ТЕРРИТОРИАЛЬНОГО КАЗНАЧЕЙСТВА**

Результат усилий по разработке программного обеспечения состоит в передаче в эксплуатацию программного продукта, удовлетворяющего требованиям пользователей. Соответственно, в процессе эксплуатации продукт будет изменяться или эволюционировать. Сопровождение программного продукта заключается в первую очередь в удовлетворении потребностей пользователя, выявление и устранение обнаруженных дефектов и ошибок, введение новых функций и компонентов в программный продукт, анализ состояния и корректировка документации.

Для сопровождения программного комплекса требуются следующие знания:

- основные понятия и предметная область сферы деятельности;
- общая характеристика, функциональные возможности, основные требования программного обеспечения;
- понимание принципа взаимодействия пользователя с территориальным казначейством.

Целью автоматизации рабочего места является повышение эффективности и качества сопровождения программного обеспечения



в процессе решения поступающих вопросов от пользователей программного комплекса.

Для достижения поставленной цели вся информация необходимая для осуществления сопровождения программного комплекса сосредоточена в одном ПО, что позволяет значительно сократить время поиска. Реализовано ведение ряда информационных справочников, используемых для оперативного отображения информации. Организация учета поступающих звонков позволяет получить статистические данные, выявить наиболее часто возникающие вопросы и, проанализировав полученную информацию, можно принимать решение о модификации недостатков программного комплекса или нахождении оптимальных путей в удовлетворении потребностей пользователя.

В настоящее время разработанное программное обеспечение находится на стадии эксплуатационного тестирования на рабочем месте инженера по сопровождению задачи «Клиент территориального казначейства» в Главном управлении Министерства финансов по Гомельской области.

**М.В. Яцковец (УО «ГГУ им. Ф. Скорины», Гомель)**  
Науч. рук. **Г.Л. Карасёва**, канд. физ.-мат. наук, доцент

## **СОЗДАНИЕ БАЗЫ ДАННЫХ АВТОСЕРВИСА**

Одной из острых проблем предприятий являются невысокие оперативность и эффективность управления товаропроводящими сетями, что обусловлено недостаточным использованием информационных технологий. При осуществлении коммерческой деятельности в автосервисе необходимо вести учет товара, как для быстрого получения информации о каком-либо наименовании, так и для ведения отчетности и статистики. Внедрение в автосервис программного обеспечения позволяет повысить эффективность работы пользователя и снизить количество рабочего времени. Ядром любой информационной системы является база данных. Хранение данных, необходимых работникам организации, в базах обладает неоспоримым преимуществом. Такие данные легче контролировать, согласовывать, изменять. Ускоряется их поиск, обработка, анализ и обеспечивается надежная защита. Поэтому специалисты автосервисов должны владеть технологиями работ с базами данных. В частности, уметь их наполнять данными; формировать запросы к базе; создавать формы для удобной работы и отчеты для анализа информации; опубликовывать данные на Web-страницах, обеспечивающих удаленный

доступ к нужной информации; автоматизировать работу с базой данных. Учитывая широкое распространение компьютерных сетей, организованных по модели «клиент/сервер», востребованным становится умение работы с многопользовательскими базами данных в архитектуре клиент/сервер.

Разрабатываемое приложение позволяет выполнять следующие операции:

- ведение справочников клиентов, товаров на складе, заказов, архив товаров;
- заполнение данных о клиентах и (имя, номер телефона, дата визита, скидка);
- формирование отчетных документов (перечень полученных товаров; перечень клиентов; полученная прибыль и т. д.).

Система предусматривает разделение пользователей по ролям: администратор, модератор, гость. Администратору доступны все действия с ресурсом, добавление и удаление пользователей из системы, настройка доступа, отслеживание изменений. Модератор может работать с оформлением заказов. Гость может просматривать список товаров автосервиса.

**М.В. Яцковец (УО «ГГУ им. Ф. Скорины», Гомель)**

Науч. рук. **Г.Л. Карасёва**, канд. физ.-мат. наук, доцент

## **РАЗРАБОТКА WEB-САЙТА СПОРТИВНОГО КЛУБА НА ОСНОВЕ PHP**

В современном обществе, когда все время молодых людей приковано к мониторам компьютеров, одной из главных задач, которая стоит перед спортивными комплексами и учреждениями является возможность показать молодежи, что «в здоровом теле» не просто «здоровый дух». Что время, проведенное на природе в компании сверстников, приносит наслаждение и гармонию в собственный внутренний мир. Спортивные организации находятся в сложной ситуации и вынуждены заниматься активным поиском людей, которым будет интересен предложенный ими вид спорта.

Для рекламных целей создаются информационные сайты, на которых можно приобрести всю необходимую информацию, а также задать свои вопросы.

Данная курсовая работа посвящена разработке web-сайта для одного из таких спортивных учреждений, а именно – для спортивного клуба смешанных единоборств «Легион».

Для разработки web-сайта спортивного клуба «Легион» использовалась связка HTML+PHP+MySQL в несколько раз превосходящая обычные HTML-сайты по функциональности, удобству и т. д.

PHP – один из немногих языков программирования, созданных специально для разработки веб-приложений. MySQL – компактный многопоточный сервер баз данных. MySQL характеризуется большой скоростью, устойчивостью и легкостью в использовании.

Синтаксис PHP очень похож на синтаксис C или Perl. Люди, знакомые с программированием, очень быстро смогут начать писать программы на PHP. В этом языке нет строгой типизации данных и нет необходимости в действиях по выделению/освобождению памяти.

Программы, написанные на PHP, достаточно легко читаемы. Написанный PHP – код легко зрительно прочитать и понять, в отличие от Perl-программ.

Конечной целью курсовой работы было создание небольшого сайта-визитки, наиболее полно характеризующего деятельность спортивного клуба «Легион». Были рассмотрены и изучены основные принципы проектирования web-приложений, необходимые для разработки сайта. Все поставленные цели были успешно выполнены и проиллюстрированы в курсовой работе.

# Содержание

## Секция 3 «Автоматизация исследований»

<b>Корзубова Н.В.</b> Платформа «1С:ПРЕДПРИЯТИЕ» как средство разработки .....	3
<b>Корзубова Н.В.</b> Технологические средства разработки .....	6
<b>Котович А.А.</b> Инновационный сервис реализации автозапчастей, построенный на технологии White Label .....	7
<b>Красотин К.В.</b> Проектирование локальной сети предприятия «Гомельский завод торгового оборудования».....	10
<b>Купреева А.В.</b> Разработка приложения при помощи средств Visual FoxPro .....	13
<b>Купреева А.В.</b> Разработка программного модуля «Охрана труда» при помощи средств Visual FoxPro .....	16
<b>Кучеров А.И.</b> Пути обеспечения надежности функционирования узлов локальной вычислительной сети .....	17
<b>Лаврук П.Ю.</b> Проблема переполнения таблицы MAC-адресов в широковещательном домене сети УО «ГГУ им. Ф. Скорины» ...	20
<b>Лаврущик Ю.Н.</b> Автоматизация рабочего места продавца-консультанта и менеджера торгового зала .....	22
<b>Лаврущик Ю.Н.</b> Решения на базе 1С: РОЗНИЦА 8.....	26
<b>Леванцов Е.В.</b> Использование интегрированных медиаустройств	27
<b>Леванцов С.В.</b> Функциональные возможности сервиса Datalex travel distribution platform по онлайн заказу авиабилетов.....	30
<b>Леванцов С.В.</b> Автоматизация заказов авиабилетов с помощью системы Travel Distribution Platform .....	31
<b>Леоненко А.М.</b> Анализ средств автоматизации лицевого счета в среде 1С:Предприятие 8.....	34
<b>Леоненко А.М.</b> Автоматизация лицевого счета в среде 1С:Предприятие 8 .....	35
<b>Лещинский А.Н.</b> Система учета и контроля заявок в Гомельском филиале РУП «Белпочта».....	39
<b>Лещинский А.Н.</b> Необходимость автоматизации процесса учета и контроля заявок в Гомельском филиале РУП «Белпочта».....	41
<b>Лишко Ю.В.</b> Вопросы реализации мультипроцессинговой системы .....	42
<b>Лосев А.А.</b> Модернизация ЛВС ОАО «Гомельстекло» .....	43
<b>Лосев А.А.</b> Проект модернизации ЛВС здания заводоуправления ОАО «Гомельстекло» .....	46
<b>Лотыш Д.А.</b> Применение графена .....	47

<b>Мамичев И.С.</b> Определение параметров эмпирической формулы	50
<b>Марсиков П.А.</b> Использование системы электронного документо оборота в IT-компаниях .....	51
<b>Мартинovich А.А.</b> Программирование на языке PHP. Разработка web-сайта – «Михи» .....	52
<b>Мельников Н.П.</b> Разработка проекта отображения динамиче- ской информации приемной комиссии университета .....	54
<b>Мельников Н.П.</b> Миграция данных из электронных таблиц Microsoft Excel с помощью сторонних API .....	57
<b>Мельников А.В.</b> Использование системы Matlab для решения задач теплопроводности .....	58
<b>Мельченко Е.С.</b> Проект автоматизации учета работы сервисного центра .....	60
<b>Меньшаков П.А.</b> Средство голосовой идентификации пользова- теля .....	61
<b>Мигун Д.В.</b> Устройства отображения информации с механиче- ской развёрткой .....	64
<b>Мигун Д.В.</b> Механическая развёртка для отображения информа- ции .....	65
<b>Мисоченко К.О.</b> Разработка приложения для обработки данных по регрессионному тестированию .....	68
<b>Мисоченко К.О.</b> Автоматизация рутинных задач на примере разработки приложения для построения отчетов по статистике регрессионного тестирования .....	69
<b>Михно А.В.</b> Разработка мультимедийного ресурса по дисци- плине «Организация производства и управление предприятием»	70
<b>Мкртычян А.С.</b> Накопители информации на основе ДНК .....	71
<b>Муха В.В.</b> Энергоэффективная топологическая память TRAM ....	72
<b>Навныко Н.А.</b> Разработка веб-приложения для рекрутинга со- трудников на Django Framework .....	75
<b>Навныко Н.А.</b> Архитектура веб-приложения для рекрутинга со- трудников .....	78
<b>Никольшин Д.А.</b> Преимущества использования Angularjs перед JavaScript .....	80
<b>Новиков В.С.</b> LI-FI – новая система передачи данных .....	81
<b>Осипенко В. В.</b> Обзор возможностей DIRECTX 12.....	84
<b>Осипенко В. В.</b> Новая версия DIRECTX.....	88
<b>Панасюк Р.С.</b> Автоматизация системы отчетности по интернет рекламе .....	89
<b>Панасюк Р.С.</b> Использование принципа REST для организации обмена данными клиента и сервера в приложении автоматизации отчетности интернет-рекламы .....	91

<b>Пархамович В.М.</b> Использование АРМ для автоматизации учета услуг на предприятии .....	92
<b>Пархамович В.М.</b> Автоматизация учета услуг на предприятии .....	95
<b>Петрова В.Э.</b> Средства автоматизации информационного моделирования предприятий .....	96
<b>Пилипейко М.Н.</b> Проект исследование метода обратной трассировки лучей .....	97
<b>Попков К.В</b> Применение утилиты SUDO.....	100
<b>Потарина А.В.</b> Новые возможности 1С:Предприятия версии 8.2 .....	103
<b>Потарина А.В.</b> Особенности клиентов 1С:Предприятия версии 8.2 .....	106
<b>Пранович П.М.</b> Анализ и поиск решений модернизации ЛВС в Могилёвском филиале РУП «Белтелеком» Бобруйский узел электросвязи .....	107
<b>Пышненко В.И.</b> Модель качества сетей голосовой связи на основе оборудования CISCO .....	108
<b>Рагин В.И.</b> Сравнение и реализация технологий загрузки клиентских станций .....	111
<b>Рагин В.И.</b> Проект бездисковой загрузки клиентских станций в сегменте ЛВС кафедры АСОИ .....	114
<b>Ракицкий А.А.</b> Тестирование антивирусного программного обеспечения .....	115
<b>Родинченко П.С.</b> Реализация подсистемы учета сжиженного газа .....	118
<b>Родинченко П.С.</b> Автоматизация учета сжиженного газа предприятия УП «Минскгаз».....	120
<b>Рубин В.А.</b> Визуализация поисковых запросов .....	124
<b>Рубин В.А.</b> Асинхронные задания в DJANGO для взаимодействия с API сторонних сервисов .....	127
<b>Руденко А.Е.</b> Фреймворк CORE DATA для операционной системы IOS .....	128
<b>Руденко А.Е.</b> Использование шаблона проектирования «ПУЛЛ ОБЪЕКТОВ» в игровом приложении .....	129
<b>Рушнов Е.А.</b> Оптическая система дисперсионного многоцелевого спектрометра .....	132
<b>Рыбакова А.В.</b> Автоматизация формирования конструкторской документации на детали типа «ВТУЛКА» в AUTOCAD .....	135
<b>Рябиков Ф.С.</b> Основные правила съемки фотографий для создания 3D-панорам .....	138
<b>Рябиков Ф.С.</b> Последовательность действий при создании 3D-панорамы.....	139
<b>Савченко И.Г.</b> Разработка информационной системы оптово-розничного звена торговой организации .....	142
<b>Седых Н.С.</b> SEO-анализ сайта УО «ГГУ им. Ф. Скорины».....	143

<b>Секацкий А.А.</b> Критерии выбора CMS .....	146
<b>Секацкий А.А.</b> Значение интернета и web-сайта для предприятия как маркетинговый ход .....	148
<b>Сериков Е.А.</b> Автоматизация управления рекламой на платежно-справочных терминалах РУП «Белпочта» .....	151
<b>Симанкова Е.А.</b> Графическое исследование и классификация по типу проекций отображения панорамных фотографий .....	153
<b>Симанкова Е.А.</b> Эффект параллакса при определении нодальной точки оптической системы .....	156
<b>Скаковский Е.В.</b> Разработка проекта контроля температуры и управления задвижками на базе контроллеров SIEMENS .....	157
<b>Скринникова Т.С.</b> Сравнительный анализ серверных СУБД .....	158
<b>Скринникова Т.С.</b> Новинка от GOOGLE – модульный смартфон GOOGLE ARA PROJECT .....	161
<b>Сметанко А.С.</b> Организация HTTP запросов с помощью библиотеки VOLLEY .....	163
<b>Старовойтов В.В.</b> Разработка мобильного приложения для осуществления торговых операций .....	165
<b>Стасев Д.Е., Лисун А.Е.</b> Разработка и реализация основных форм и запросов базы данных «Расчет припусков на механическую обработку» .....	166
<b>Стельченко В.Н.</b> Описание возможностей системы управления сайтом JOOMLA .....	169
<b>Стельченко В.Н.</b> Применение интернет-технологий в бизнесе.....	171
<b>Сюськин А.В.</b> Автоматизация торгового объекта под управлением системы 1С:Предприятие .....	172
<b>Сюськин А.В.</b> Необходимость автоматизации торгового объекта .	174
<b>Тарачешников Н.Г.</b> Автоматизация расчета налоговых средств в среде 1С: Предприятие 8.1 .....	175
<b>Тарачешников Н.Г.</b> Автоматизация проведения начислений работникам организации в среде 1С:Предприятие 8.2.....	177
<b>Ткачев О.М.</b> Разработка мобильного приложения «ВИКТОРИНА» для платформы IOS .....	180
<b>Тотиков В.С.</b> Автоматизация производственной деятельности склада автозапчастей ООО «ШИМ-БИ ТРАК» в системе программ 1С:Предприятие 8.....	181
<b>Тотиков В.С.</b> Автоматизация магазина автозапчастей в системе программ 1С:Предприятие 8. Основные возможности данной системы и области её применения.....	182
<b>Усольцев Г.А.</b> Создание веб-приложений на платформе GRAILS	185
<b>Филонюк Т.В.</b> Автоматизация в складской логистике – важное конкурентное преимущество .....	187

<b>Філіпчык А.В.</b> Алгарытм транскрыбавання лексічных адзінак у лінгвістычнай інфармацыйна-пошукавай сістэме .....	190
<b>Фомин Н.С.</b> Использование средств MySQL и PHP при создании баз данных .....	192
<b>Хазанова Н.В.</b> Разработка структуры базы данных учета успеваемости учащихся .....	194
<b>Хазанова Н.В.</b> Использование web-приложения для автоматизации учета успеваемости учащихся колледжа .....	197
<b>Хачетлов А.Ю.</b> Автоматизация заполнения документа «Отчёт о розничных продажах» в 1С:Предприятие .....	199
<b>Хачетлов А.Ю.</b> Важность автоматизации в деятельности предприятия .....	200
<b>Хобня А.И.</b> Декомпозиция сетей нового поколения .....	201
<b>Хобня А.И.</b> Разработка универсальной платформы моделирования механизмов обеспечения качества обслуживания в сетях пакетной передачи данных .....	204
<b>Хомицкий Ю.В.</b> Автоматизация складского учета .....	206
<b>Хомицкий Ю.В.</b> Автоматизация учета материалов на складе ...	207
<b>Храбров Д.Е.</b> Методика локального WI-FI позиционирования на основании алгоритма сопоставления с образцом .....	210
<b>Цябус А.Н.</b> Использование Unity3d для создания игрового приложения .....	213
<b>Цябус А.Н.</b> Разработка сайта на CMS Joomla с использованием расширений .....	214
<b>Черёмухин М.В.</b> 2С:Управление торговлей .....	215
<b>Чернышев А.И.</b> Основные возможности системы «1С:Предприятие 8.2» и области её применения .....	218
<b>Чернышев А.И.</b> Автоматизация сервисного центра в среде разработки «1С:Предприятие 8.2» .....	219
<b>Чигринец О.В.</b> Использование Case-средств в разработке автоматизированных информационных систем .....	222
<b>Чирик Д.В.</b> Анализ локальной вычислительной сети ОВД Речицкого райисполкома .....	225
<b>Чирик Д.В.</b> Разработка проекта модернизации локальной вычислительной сети отдела внутренних дел Речицкого райисполкома...	225
<b>Шангытбаева Г.А.</b> Влияние атак на отказ в обслуживании на компьютерные сети .....	227
<b>Шангытбаева Г.А.</b> Разработка средств защиты инфокоммуникаций на основе ПЛИС .....	229
<b>Шараев А.О.</b> Тенденции развития сетей масштаба города и предоставляемых ими сервисов .....	231



<b>Шестаков И.С.</b> Front-End и Back-End разработка приложения по учету исключительных ситуаций в информационной системе...	234
<b>Шестаков И.С.</b> Использование Java Beans при создании web-приложения по учету исключительных ситуаций в информационной системе .....	235
<b>Шинкарёва В.А.</b> Система управления базами данных Cubrid .....	236
<b>Шинкарёва В.А.</b> Сравнительная характеристика анонимных сетей I2P и TOR .....	239
<b>Штукарь А.Н.</b> Автоматизированный расчет рациональных режимов резания на плоскошлифовальных станках .....	242
<b>Юницкий Я.А.</b> Применение технологии Nvidia Cuda для параллельных вычислений .....	244
<b>Юницкий Я.А.</b> Написание приложения идентификации процессора с помощью инструкции CPUID .....	246
<b>Якубова А.С.</b> Сферические панорамы музея «Лёс салдата» г. Рогачева.....	248
<b>Якубова А.С.</b> Программное обеспечение для сборки панорамных изображений .....	251
<b>Якубова С.С.</b> Получение изображений для фотопанорам .....	254
<b>Якубова С.С.</b> Виртуальный тур по зимнему саду .....	256
<b>Ярошенко Д.А.</b> Разработка инструментария по управлению СУБД DB2 в среде ОС LINUX .....	259
<b>Яцкова Т.Н.</b> Разработка проекта модернизации информационной системы учета клиентов территориального казначейства по Гомельской области .....	261
<b>Яцкова Т.Н.</b> Автоматизация рабочего места по сопровождению программного комплекса учета клиентов территориального казначейства .....	264
<b>Яцковец М.В.</b> Создание базы данных автосервиса .....	265
<b>Яцковец М.В.</b> Разработка web-сайта спортивного клуба на основе PHP .....	266

*Научное издание*

**АКТУАЛЬНЫЕ ВОПРОСЫ  
ФИЗИКИ И ТЕХНИКИ**

Материалы IV Республиканской  
научной конференции  
студентов, магистрантов и аспирантов

(Гомель, 15 апреля 2015 года)

В трёх частях

Часть 3

Подписано в печать 29.09.2015. Формат 60×84 1/16.  
Бумага офсетная. Ризография. Усл. печ. л. 16,0.  
Уч.-изд. л. 17,5. Тираж 8 экз. Заказ 571.

Издатель и полиграфическое исполнение:  
учреждение образования  
«Гомельский государственный университет  
имени Франциска Скорины».

Свидетельство о государственной регистрации издателя, изготовителя,  
распространителя печатных изданий № 1/87 от 18.11.2013.  
Специальное разрешение (лицензия) № 02330 / 450 от 18.12.2013.  
Ул. Советская, 104, 246019, Гомель.

