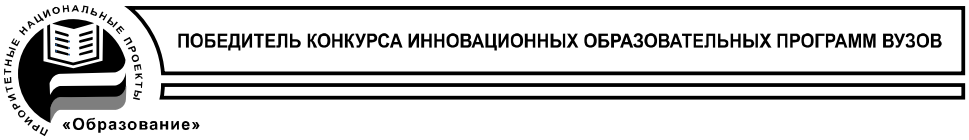
**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ**




# НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК

## Выпуск 40

# НАУЧНАЯ ШКОЛА «ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ, ПРОЕКТИРОВАНИЕ, ТЕХНОЛОГИЯ ЭЛЕМЕНТОВ И УЗЛОВ КОМПЬЮТЕРНЫХ СИСТЕМ»

## Труды молодых ученых


**САНКТ-ПЕТЕРБУРГ**
**2007**

Выпуск содержит материалы **IV межвузовской конференции молодых ученых**., организованной 10–13 апреля 2007 года Санкт-Петербургским государственным университетом информационных технологий, механики и оптики в сотрудничестве с
Балтийским государственным техническим университетом «Военмех»
Башкирским государственным университетом
Белорусским государственным педагогическим университетом им. Максима Танка
Белорусским государственным технологическим университетом
Белорусским государственным университетом информатики и радиоэлектроники
Дальневосточным государственным университетом
Дальневосточной академией государственной службы
Институтом аналитического приборостроения Российской Академии Наук (РАН)
Институтом Солнечно-Земной Физики СО РАН
Институтом химии высокочистых веществ РАН (г. Нижний Новгород)
Казанским государственным техническим университетом им. А.Н. Туполева
Казанским государственным университетом
Карельским государственным педагогическим университетом
Костромским государственным технологическим университетом
Красноярским государственным техническим университетом
Ленинградским государственным университетом им. А.С. Пушкина
Магнитогорским государственным техническим университетом им. Г.И. Носова
Морской государственной академией им. адмирала Ф.Ф. Ушакова
Московским государственным институтом электронной техники (техническим университетом)
Московским государственным техническим университетом им. Н.Э. Баумана
Московским педагогическим государственным университетом
Муромским институтом Владимирского государственного университета
Петербургским государственным университетом путей сообщения
Пятигорским государственным лингвистическим университетом
Российским государственным гидрометеорологическим университетом
Самарским государственным архитектурно-строительным университетом
Санкт-Петербургским государственным горным институтом им. Г.В. Плеханова (техническим университетом)
Санкт-Петербургским государственным инженерно-экономическим университетом (ИНЖЭКОН)
Санкт-Петербургским государственным политехническим университетом
Санкт-Петербургским государственным университетом
Санкт-Петербургским государственным университетом аэрокосмического приборостроения
Санкт-Петербургским институтом машиностроения (ЛМЗ-ВТУЗ)
Санкт-Петербургским университетом кино и телевидения
Санкт-Петербургской государственной академией физической культуры им. П.Ф. Лесгафта
Санкт-Петербургской государственной педиатрической медицинской академией
Северо-Западной академией государственной службы
Северо-Осетинским государственным университетом им. К.Л. Хетагурова
Тамбовским государственным университетом им. Г.Р. Державина
Татарским государственным гуманитарно-педагогическим университетом
Университетом Aix-Marseille II (Франция)
Университетом Прованса (Франция)
ФГУП ″ЦНИИ им. академика А.Н. Крылова″
Энгельсским технологическим институтом Саратовского государственного технического университета

В выпуске представлены работы, поддержанные финансированием в рамках:

– аналитической ведомственной целевой программы «Развитие научного потенциала высшей школы (2006–2008 гг.) (Федеральное агентство по образованию);

– Федеральной целевой программы развития образования на 2006–2010 гг. (Федеральное агентство по образованию);

– Российского фонда фундаментальных исследований,

а также инициативные разработки.

## ПРОГРАММНЫЙ КОМИТЕТ КОНФЕРЕНЦИИ

**Председатель** – ректор СПбГУ ИТМО, д.т.н., профессор **В.Н. Васильев**
**Сопредседатели** – проректор по развитию, д.т.н., профессор **В.О. Никифоров,**
проректор по УО и АР, д.ф.-м.н., профессор **Ю.Л. Колесников,**
проректор по УМР, к.т.н., профессор **А.А. Шехонин,**
декан факультета ППО, д.т.н., профессор **В.Л. Ткалич**
**Члены программного комитета** – д.т.н., профессор **Ю.А. Гатчин**, д.т.н., профессор **В.М. Мусалимов**, д.т.н., профессор **С.Б. Смирнов**, д.т.н., профессор **В.А. Тарлыков**, д.т.н., профессор **Е.Б. Яковлев**, к.т.н. **Т.В. Точилина**

## ОРГАНИЗАЦИОННЫЙ КОМИТЕТ КОНФЕРЕНЦИИ

**Председатель –** начальник НИЧ **Л.М. Студеникин**
**Зам. председателя –** к.т.н. **Т.В. Точилина**
**Члены организационного комитета – П.А. Борисов, Н.Н. Валентик, И.Н. Жданов, С.Ю. Керпелева, Н.В. Когай, А.В. Козаченко, И.М. Кудрявцева, Д.В. Лукичёв, А.А. Малинин, Л.В. Можжухина, Ю.С. Монахов, Н.Б. Нечаева, М.В. Никитина, М.С. Петрищев, С.С. Резников, В.Н. Фролков**


В 2007 году СПбГУ ИТМО стал победителем конкурса инновационных образовательных программ вузов России на 2007–2008 годы. Реализация инновационной образовательной программы «Инновационная система подготовки специалистов нового поколения в области информационных и оптических технологий» позволит выйти на качественно новый уровень подготовки выпускников и удовлетворить возрастающий спрос на специалистов в информационной, оптической и других высокотехнологичных отраслях экономики.

**ISSN 1819-222X**

# 1 СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ

## АВТОМАТИЗАЦИЯ ТЕПЛОВЫХ РАСЧЕТОВ ЭЛЕКТРОННЫХ БЛОКОВ С ПОМОЩЬЮ САПР SOLIDWORKS/COSMOSWORKS НА ЭТАПЕ КОНСТРУКТОРСКОГО ПРОЕКТИРОВАНИЯ

**Д.А. Боголюбов, Н.С. Григорьева*, О.В. Елисеев, Н.В. Когай**
*** (ОАО «Российский институт радионавигации и времени»)**
**Научный руководитель – к.т.н., доцент Н.С. Кармановский**

В статье освещается проблема автоматизации тепловых расчетов на предприятии с помощью САПР. Приведены результаты исследования блоков на работоспособность по тепловым режимам, а также выработанные рекомендации по созданию трехмерных моделей.

Данная статья подготовлена в рамках проекта по внедрению САПР SolidWorks в Российском институте радионавигации и времени. Актуальность работы связана с практикой автоматизации различных расчетов, в том числе тепловых. До проведения настоящего проекта расчеты производились вручную, что требовало существенных затрат времени и не всегда гарантировало требуемую точность.

Был произведен анализ существующих САПР, на основании которого выявлено, что использованная система отличается большой функциональностью, а также позволяет исполнять чертежи в соответствии с ЕСКД (табл. 1).

| CAD/CAE система | Чертежи по ЕСКД | Тепловой расчет | Расчет механических напряжений | Расчет на ударную нагрузку | Автоматическая корректировка конструкции |
|---|---|---|---|---|---|
| SolidWorks | + | + | + | + | + |
| AutoCAD | + | – | – | – | – |
| MentorGraphics | – | + | + | + | + |
| ANSYS | – | + | + | + | + |

Таблица 1. Функциональность САПР

После проведения анализа для автоматизации тепловых расчетов электронных блоков была выбрана САПР SolidWorks версии 2006 года. SolidWorks включает в себя конструкторскую и расчетную часть, а также вспомогательные программы.

Одной из важнейших составляющих САПР SolidWorks является программа инженерного анализа COSMOSWorks. Она предоставляет большое количество встроенных инструментов проектирования и расчета, в том числе: тепловой расчет; расчет на ударную нагрузку; расчет механических напряжений и деформаций; расчет собственных частот; автоматическая корректировка конструкции модели и другие расчеты.

Тепловые расчеты в COSMOSWorks основаны на методе конечных элементов. Используемый алгоритм фронтального исключения позволяет экономить оперативную память персонального компьютера и время расчета. Это реализовано за счет последовательного включения элементов в матрицу фронта [1].

Исследовавшиеся конструктивы являются типичными для данного предприятия и иллюстрируют различные виды теплоотвода. Все конструктивы относятся к бортовой аппаратуре, применяемой в глобальной навигационной спутниковой системе ГЛОНАСС. При подготовке работы они были классифицированы по назначению и типу теплоотвода.

После анализа алгоритмов теплового расчета были выработаны требования к исходным данным.

1. Необходимо задать в качестве начальных условий начальную температуру элементов. Особенно это важно для базового крепежного элемента, через который осуществляется кондуктивный теплообмен. Необходим, кроме того, тщательный учет теплоотводов, крайне важно учесть термостатируемые элементы.

2. Так как анализ тепловых режимов в COSMOSWorks 2006 и выше позволяет учитывать все виды теплообмена (конвективный, кондуктивный и излучение), необходимо определить наиболее значимые составляющие и в качестве исходных данных задавать только их с целью экономии затрат мощностей компьютера.

3. С большой осторожностью следует подходить к заданию коэффициентов теплопередачи, так как расчет контактных тепловых сопротивлений не всегда происходит корректно и требует существенных затрат памяти. При этом необходим учет тепловых контактов в некоторых принципиальных узлах.

4. Необходимо учитывать, что при измельчении конечно-элементной сетки значения некоторых параметров могут быть теоретически бесконечными вследствие особенностей внутренних алгоритмов программы COSMOSWorks. В расчете тепловых режимов это относится, прежде всего, к параметру «Тепловая мощность». При этом следует понимать, что от параметров конечно-элементной сетки зависит ход всего расчета в целом. Кроме того, чрезмерное уточнение исходных данных приводит к увеличению размерности задачи, вследствие чего программе может не хватить памяти, и расчет будет прекращен.

5. Конечно-элементная модель задачи теплопроводности предполагает, что в каждом узле присутствует только одна степень свободы – температура [2].

Были исследованы следующие конкструктивы.

1. Рубидиевый стандарт частоты (рис. 1) представляет собой несколько тепловыделяющих элементов, установленных на термостатируемой плите. Ввиду чрезмерной перегруженности деталями исходной трехмерной модели была создана тепловая модель. Результат дискретизации для тепловой модели стандарта частоты представлен на рис. 2. Построение сетки на такой модели прошла без затруднений.

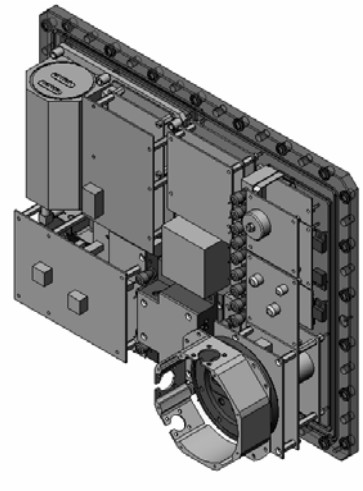


Рис. 1. Трехмерная модель рубидиевого стандарта частоты

Тепловой расчет на такой трехмерной модели прошел без затруднений (рис. 3).

Стандарт частоты признан работоспособным по тепловым режимам. Погрешность методики расчета составила 1,8 %. Градиент температур, как видно из шкалы, не превысил 1,3 К. Предполагаемое время расчета данного блока вручную составило около 4 часов, расчет тепловых режимов с помощью САПР (включая построение тепловой модели) – 3 часа.

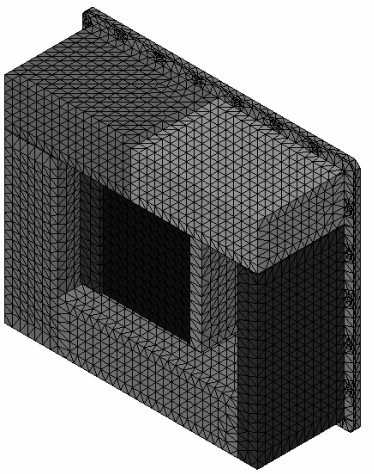


Рис. 2. Конечно-элементная модель рубидиевого стандарта частоты

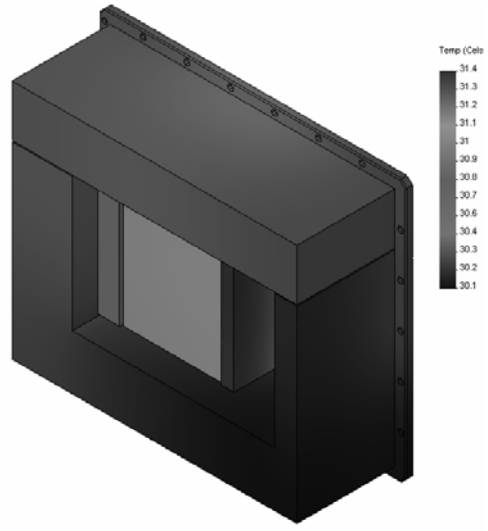



Рис. 3. Результаты теплового расчета рубидиевого стандарта частоты

2. Дискриминатор стандарта частоты (рис. 4). Теплоотвод осуществляется через термостатируемую плиту. Особенность этой модели состоит в моделировании эффекта Пельтье, для которого нет специальной функции в программе. Элемент Пельтье был задан как многослойный элемент, причем для каждого слоя задавалась сверхмалая либо сверхбольшая теплопроводность. В целом исходная модель достаточно упрощена, она была использована и для расчета, тепловая модель не строилась.

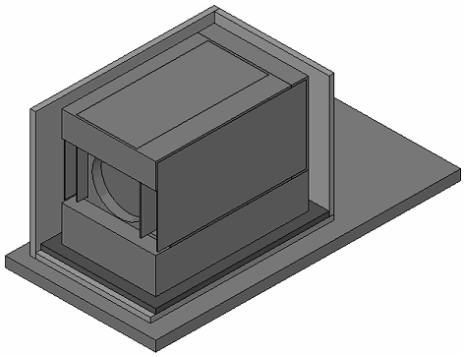


Рис. 4. Трехмерная модель дискриминатора стандарта частоты

Тепловой расчет на такой модели прошел без затруднений, несмотря на определенные сложности при конечно-элементной дискретизации плиты со встроенным элементом Пельтье. Результаты расчета представлены на рис. 5.

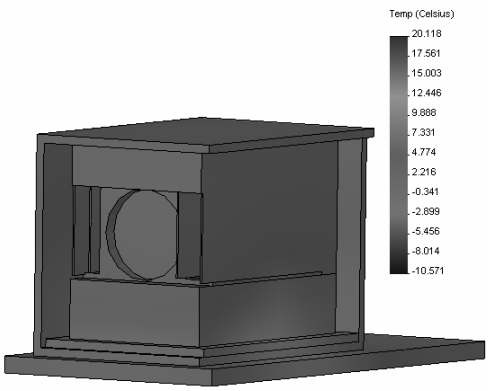



Рис. 5. Результаты теплового расчета дискриминатора

Дискриминатор признан неработоспособным по тепловым режимам вследствие возникновения большого градиента температур. Отмечена высокая погрешность расчета. Очевидно, это связано с неудачным моделированием эффекта Пельтье. Для исследования таких элементов предлагается использовать узкоспециализированные программные пакеты.

При составлении протокола испытаний результат данного автоматизированного расчета не принимался во внимание.

3. Измерительный модуль. Трехмерная модель измерительного модуля представлена на рис. 7. Размещенный на плате термостабилизирующий элемент должен за требуемое время нагреть прочие элементы конструкции до рабочей температуры. В результате расчета было выяснено, что требуется изменить расположение тепловыделяющих элементов на плате для более равномерного распределения температуры. Измененная конструкция представлена на рис. 8. Было увеличено число нагревательных элементов одновременно с уменьшением тепловой мощности каждого из них в отдельности.

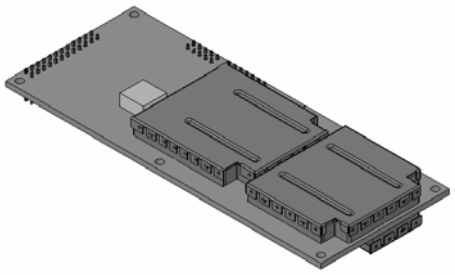


Рис. 6. Трехмерная модель измерительного модуля

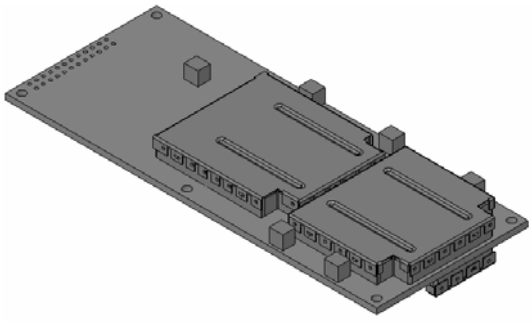


Рис. 7. Трехмерная модель измерительного модуля в измененном конструктиве

Измерительный модуль в измененном конструктиве признан работоспособным по тепловым режимам. Градиент температур составил 6 К. Такие значения не превышают допустимых.

После анализа затраченного на расчеты времени были сделаны выводы о целесообразности автоматизации тепловых расчетов с помощью данной САПР. Эти выводы приведены в табл. 2.

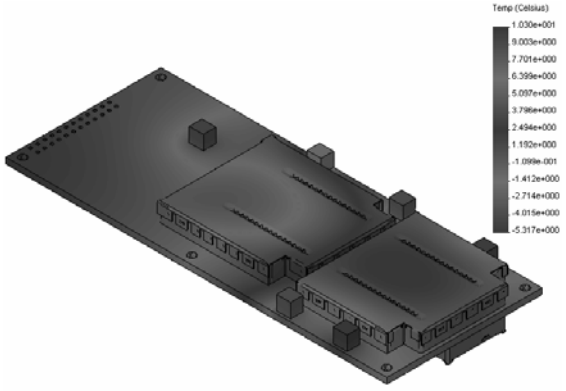



Рис. 8. Результаты теплового расчета измерительного модуля

| Параметр | Стандарт частоты | | Дискриминатор | | Измерительный модуль | |
|---|---|---|---|---|---|---|
| | вручную | САПР | вручную | САПР | вручную | САПР |
| Время расчета, час. | 6 | 4 | 3 | 1,5 | 4 | 1,5 |
| Трудоемкость, чел.-ч. | 8 | 7 | 5 | 3 | 6 | 3 |

Таблица 2. Результаты автоматизации тепловых расчетов с помощью САПР SolidWorks

Видно, что автоматизация тепловых расчетов увеличивает экономическую эффективность процесса. Трудоемкость автоматизированного расчета в среднем на 35 % меньше, чем ранее применявшегося на предприятии расчета вручную.

На основании произведенного моделирования и расчетов были выработаны следующие рекомендации по созданию трехмерных моделей:

- исключить из модели элементы со сложной геометрией;
- упростить геометрию нагревательных элементов;
- каждый отдельный элемент создавать отдельным файлом;
- задавать все возможные параметры соединений между элементами.

Также сделан вывод о невозможности применения данной САПР при расчете элементов Пельтье. Рекомендовано использовать узкоспециализированные программы.

Достигнутые результаты:

- проведена классификация конструктивов;
- проведено внедрение САПР на предприятии;
- проведен расчет тепловых режимов выбранных конструктивов;
- вынесены заключения о работоспособности блоков по тепловым режимам;
- на основании полученных результатов тепловых расчетов выработаны рекомендации по построению трехмерных моделей.

## Литература

1. Алямовский А.А. «SolidWorks/COSMOSWorks. Инженерный анализ методом конечных элементов». М.: ДМК Пресс, 2004. 432 с.
2. Алямовский А.А., Собачкин А.А., Одинцов А.А., Харитонович А.И., Пономарев Н.Б. SolidWorks. Компьютерное моделирование в инженерной практике. СПб: BHV, 2004. 306 с.

# АВТОМАТИЗАЦИЯ ПРОЦЕССА РАЗРАБОТКИ WEB-ПРИЛОЖЕНИЙ НА ПРИМЕРЕ FRAMEWORK-СИСТЕМЫ КОМПАНИИ DIGART

**В.В. Заря, А.А. Протченков, Е.В. Симаков**
**Научный руководитель – д.т.н., профессор А.Г. Коробейников**

Ускорение процесса разработки web-приложений без ущерба надежности и безопасности является одной из ключевых стратегических задач для любой компании в сфере веб-технологий. Разработка набора средств языка, выступающих в качестве каркаса для создания приложений, обеспечивает централизацию управления, реализацию единого стандарта разработки, упрощение развертывания, что в значительной степени сокращает временные затраты на создание и поддержку web-приложения.

## Введение

С развитием интернет-технологий каждая фирма получила возможность создания своего официального представительства в глобальной сети для решения различных бизнес-задач. Для осуществления большинства задач требуется создание полноценных web-приложений, размещенных на стороне сервера, которые в процессе своей работы обращаются к различным ресурсам сервера, в том числе и к базам данных. Реализация многофункциональных web-приложений требует системного подхода, который предусматривает определение необходимых средств и технологий. При создании и выборе средств, а также технологий разработки главным критерием становится стоимость, скорость и качество разработки. Поэтому важной задачей в процессе создания web-приложений является создание набора средств, ускоряющих процесс разработки без ущерба надежности и безопасности.

В настоящее время существует множество framework-систем[1]. Наиболее распространенные из них – Zend Framework, CakePHP, Symfony Project, Seagull Framework, WACT, Prado, PHP on TRAX, ZooP Framework, eZ Components, CodeIgniter [1]. Основные решения, которые в них используются:

- поддержка паттерна Model-View-Controller[2];
- расширенные возможности слоя представления (Presentation layer);
- ORM[3];
- поддержка шаблонизатора;
- поддержка понятных url`s.

Для решения поставленной задачи в данной работе была спроектирована логическая архитектура framework-системы, в которой были использованы все существующие решения, а также была усовершенствована архитектура и добавлены новые технологии.

## Определение требований к framework-системе

Программирование на стороне сервера в настоящее время является необходимым условием для решения широкого спектра задач. Оно дает возможность:

- получать и обрабатывать на сервере данные, введенные пользователем при помощи формы;

[1] Framework-система − набор средств языка логически объединенных в один пакет и выступающих в качестве каркаса для создания приложений, которые в свою очередь уже решают конкретно поставленные задачи.
[2] Model-View-Controller − архитектура программного обеспечения, в которой модель данных приложения, пользовательский интерфейс и управляющая логика разделены на три отдельных компонента, так, что модификация одного из компонентов оказывает минимальное воздействие на другие компоненты.
[3] ORM − Object-Record Mapper, технология преобразования объектов в реляционную базу данных.

- динамически создавать web-документы, не зависящие ни от платформы, ни от браузера клиента;
- обеспечивать динамический доступ к данным, находящимся на сервере, в частности, к серверным базам данных;
- использовать серверные компоненты, предназначенные для решения типовых задач;
- осуществлять аутентификацию пользователя.

Поэтому обычно заранее разрабатываются базовые принципы и подходы для каждой из часто встречающихся задач. Очень часто такие решения объединяются общей программной средой, называемой framework-системой. Тем самым увеличивается скорость разработки приложений, одновременно повышается их безопасность и надежность и обеспечивается легкость сопровождения.

Перечислим задачи, которые должна решать framework-система:

- обеспечение единого стандарта разработки web-приложений;
- обеспечение адекватного реагирования приложения на непредвиденные ситуации;
- централизация управления приложениями;
- обеспечение режима быстрой разработки;
- командная работа;
- ориентация на персонал средней квалификации;
- поддержка совместной работы специалистов разных областей;
- легкость сопровождения.

На основе приведенных задач сформулированы общие требования к системе:

- поддержка паттерна Model-View-Controller;
- расширенный слой представления (Presentation layer);
- поддержка шаблонизатора;
- поддержка модульной структуры;
- инсталляция готовых модулей в систему;
- движок нефункциональных связей[4];
- многофункциональный уровень управления данными (Persistence layer), включая ORM.

Framework-система обычно состоит из следующих компонентов:

- среда исполнения приложения;
- библиотека классов и функций.

Функциями среды исполнения является запуск запрошенного приложения и организация общения между компонентами приложения и необходимыми им ресурсами framework-а. Понятно, что без среды выполнения невозможен запуск таких приложений [2]. Библиотека классов и функций включает многократно используемые компоненты, которые могут быть применены в любом месте приложения. Тем самым облегчается жизнь прикладным программистам и дает им возможность заниматься решением конкретных задач. На рис. 1 показан пример взаимодействия приложения с framework-системой. Для запуска приложения необходим некоторый сервис, задача которого – обратиться к среде исполнения с запросом запуска приложения. Далее среда определяет точку входа приложения, которым может являться компонент «модуль», инициирует его и передает ему управление. В процессе работы модуль приложения посредством среды исполнения (на рисунке это не указано) обращается к библиотеке framework-а и использует ее для решения своих задач.

В качестве примеров framework-систем могут выступать известные решения – OpenGL, CMF (Content Management Framework), .NET Framework.

[4] Под нефункциональными связями понимается связь между собой независимых модулей посредством шаблонов. Это необходимо для вывода на одной странице информации из различных модулей.

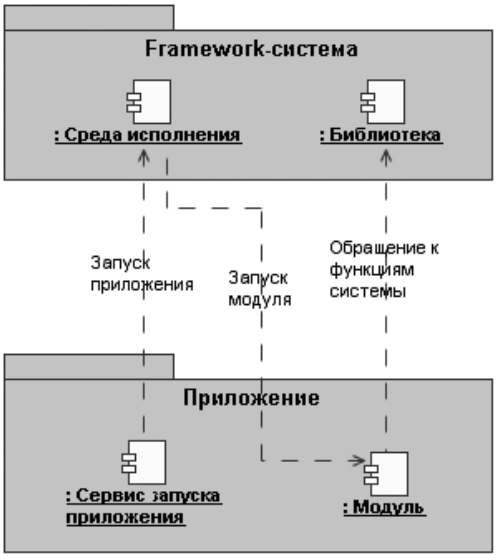



Рис. 1. Пример взаимодействия приложения и framework-системы

**Физическая архитектура системы**

На рис. 2 представлена физическая архитектура системы с точки зрения web-приложения. Точкой входа в приложение и единственной точкой взаимодействия является скрипт index.php, который обращается к набору классов каркаса приложения, находящемуся в пакете «Web-приложение». Этот пакет, в свою очередь, обращается к среде выполнения framework-системы, которая уже организует работу всего web-приложения. В ходе работы приложения может происходить обращение к необходимым ресурсам и базам данных расположенных на web-сервере.

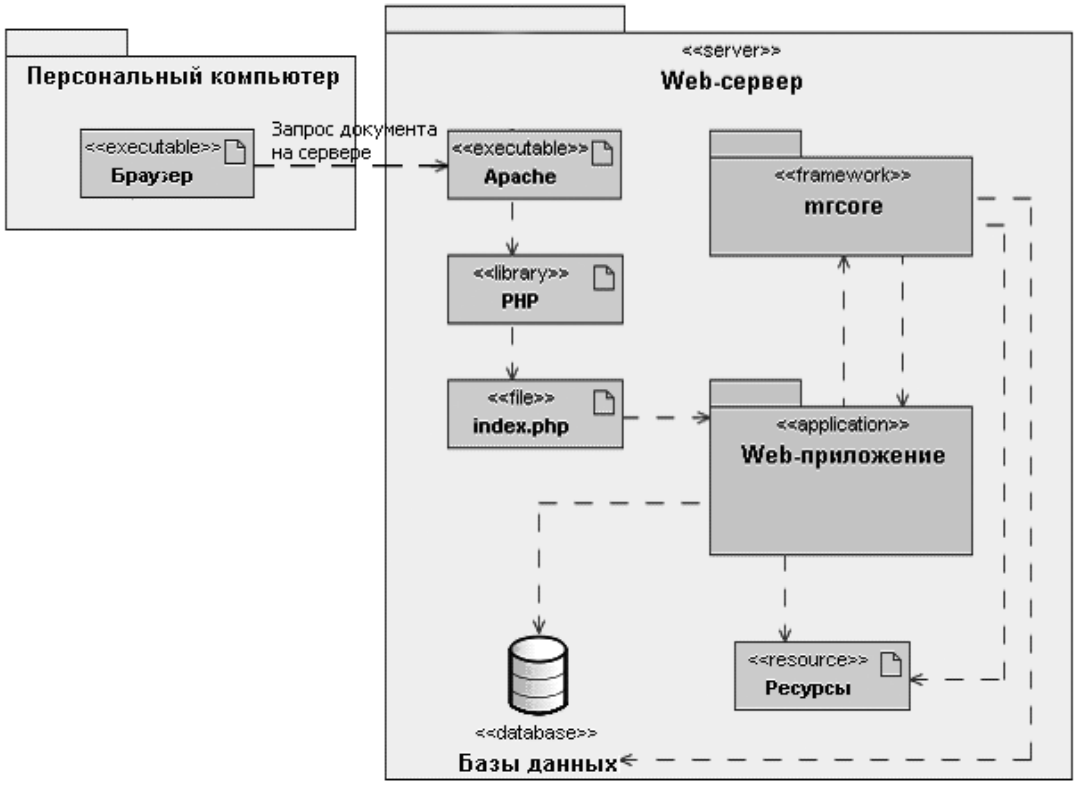



Рис. 2. Физическая архитектура системы с точки зрения web-приложения

### Логическая архитектура системы

Логическая архитектура системы с точки зрения разделения на функциональные слои представлена статической моделью абстрактной машины, изображена на рис 3. В данной архитектуре вводится три уровня абстракции:

- Presentation layer (уровень представления);
- Logical layer (уровень бизнес-логики);
- Persistence layer (уровень управления данными).

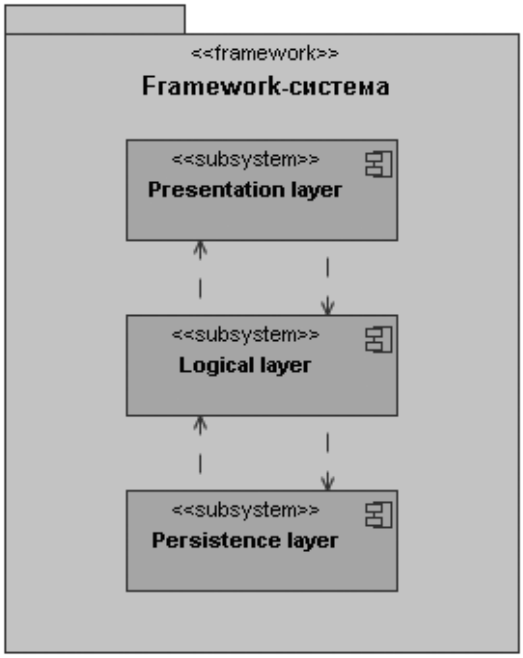



Рис. 3. Логическая архитектура системы

**Уровень представления (Presentation layer)** предназначен для вывода сформированных данных другими слоями, обычно на экран пользователю [3]. В требованиях к системе к этому слою были выдвинуты особые требования:

- высокий процент повторного использования компонентов данного слоя;
- новые компоненты должны легко встраиваться в систему;
- внутренняя логика компонентов должна быть максимально отделена от оформления, и должна быть минимально связана с другими подсистемами.

На основе этих требований подсистема была разделена на компоненты, которые приведены на рис. 4. Все классы, наследуемые от Control, являются компонентами графического web-интерфейса, которые инкапсулируют собственную логику поведения и имеют возможность представления своих данных в виде структурированного блока.

Чтобы разделить труд программиста и верстальщика, необходим шаблонизатор способный преобразовать структурированные блоки, сгенерированные визуальными компонентами, в html-код со всеми особенностями оформления. Этот код – окончательно сформированный документ по запросу клиента – будет являться результатом работы приложения, и web-сервер передаст его браузеру пользователя. Чтобы система не была привязана к определенному шаблонизатору, вводится интерфейс ITemplater. Он предназначен для стыковки постороннего шаблонизатора со слоем представления.

Сейчас в системе в качестве стороннего шаблонизатора используется Smarty. Для интеграции этого компонента в систему был создан класс SmartyAdapter. Он агрегирует в себе компонент Smarty и реализует интерфейс ITemplater, что позволяет ему выступать в роли шаблонизатора в системе.

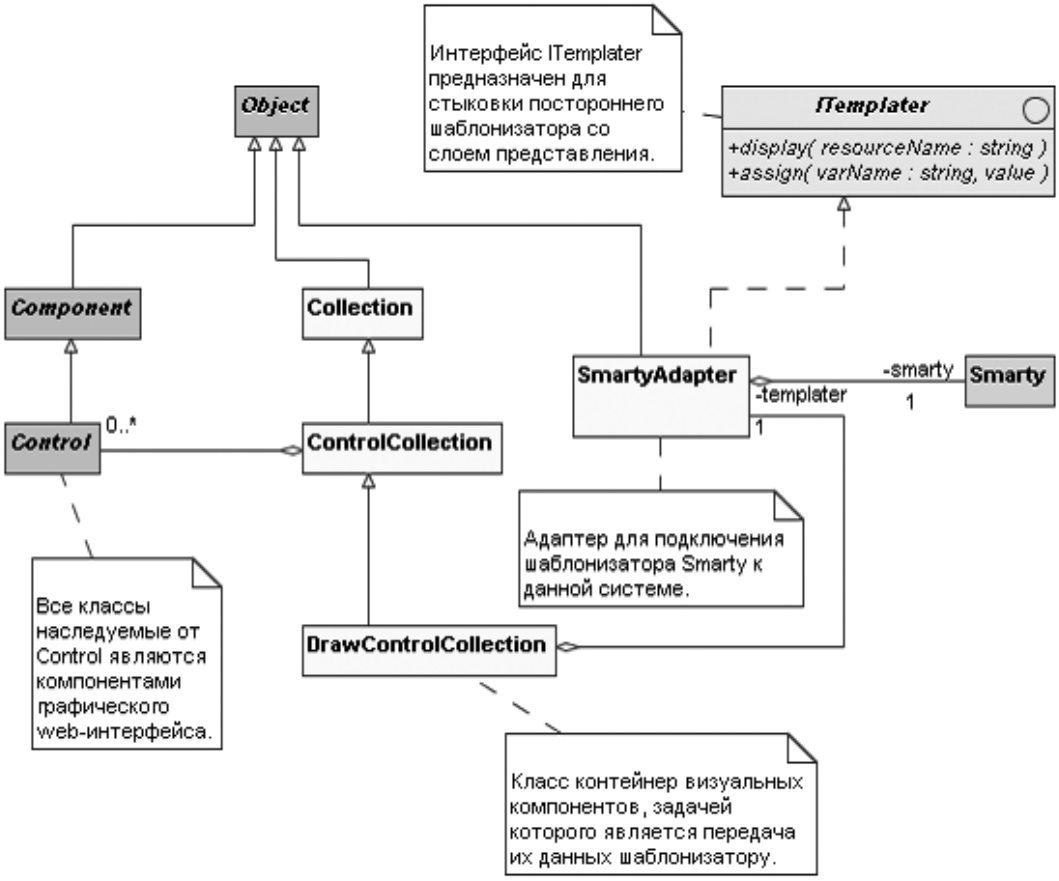



Рис. 4. Объектная модель слоя представления

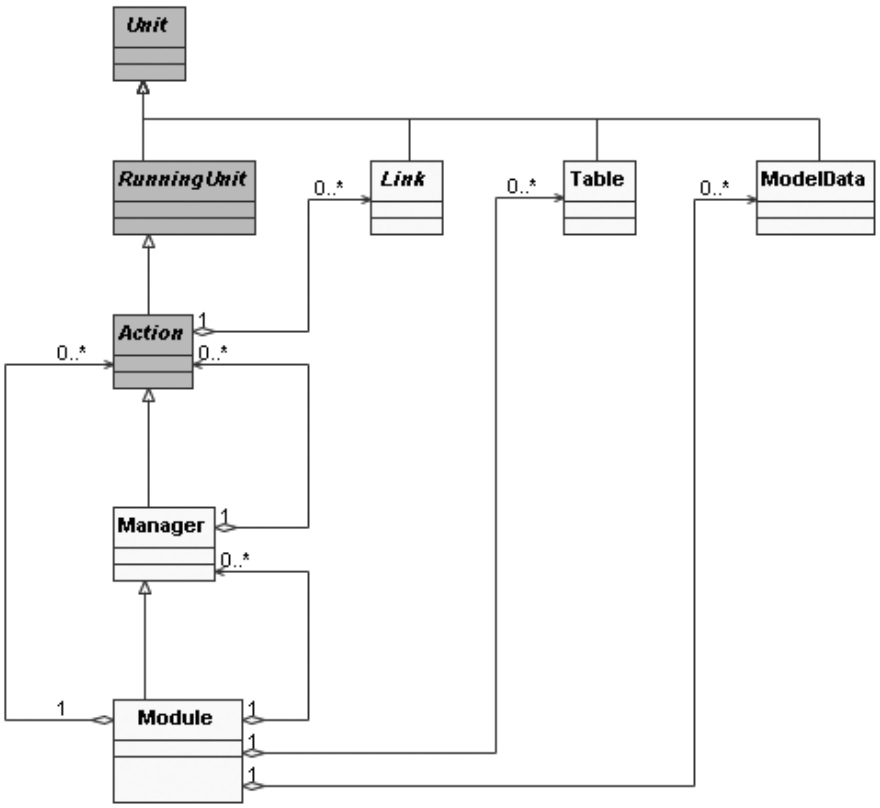



Рис. 5. Основные компоненты логического уровня

**Логический уровень (Logic layer).** Одним из предназначений логического уровня является группировка компонентов. Основные компоненты представлены на рис. 5. Для этого вводится понятие компонентов-контейнеров. Наследуются такие компоненты от класса Action. Любой объект, который был наследован от этого класса, может быть запущен по запросу клиента, при условии, что этот компонент предварительно был зарегистрирован в системной таблице запускаемых юнитов. Самостоятельной единицей из этой иерархии является класс Module. Он имеет права включать в себя все типы компонентов, наследуемых от класса Unit, кроме компонентов, имеющих тип Module. Промежуточный компонент Manager введен для удобства компоновки элементов. Вложенность менеджеров друг в друга может быть произвольной. При внешнем вызове объекта, который лежит в глубине иерархии юнитов, будут вызваны по порядку все родительские контейнеры, и только после этого управление будет передано вызываемому юниту. Такой механизм дает гарантию того, что при работе последнего по иерархии компонента уже будут инициализированы все компоненты, входящие в состав модуля, и он сможет обратиться к любому из них, если это ему разрешено политикой логического слоя.

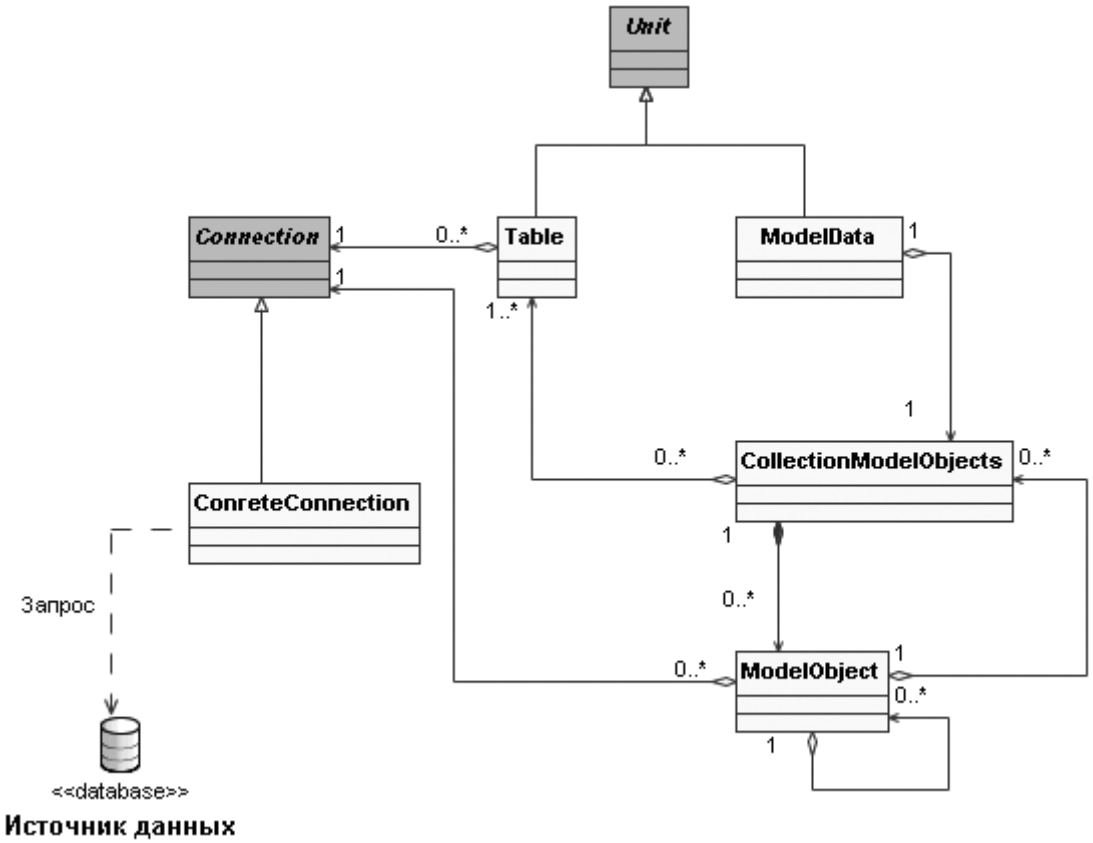



Рис. 6. Объектная модель слоя управления данными

**Уровень управления данными (Persistence layer),** объектная модель которого представлена на рис. 6, решает две задачи. Во-первых, он организует взаимодействие непосредственно с базой данных с помощью объектов Table и ConreteConnection. А, во-вторых, он реализует ORM (Object-Record-Mapper, слой преобразования объектов в реляционную базу данных) с помощью компонентов ModelData, CollectionModelObjects и ModelObject. Последний механизм (ORM) позволяет не задумываться об устройстве таблиц базы данных и работать только с модельными объектами и их коллекциями. Правда, нужно помнить, что это может сказаться на производительности. Наследники от класса ModelObject будут далее называться модельными объектами, так как они вы-

полняют роль моделей объектов, которые встречаются в реальном мире. Их главная функция заключается в помощи решения задач различных предметных областей, основываясь на объектно-ориентированном подходе.

### Механизм нефункциональных связей

Под нефункциональными связями понимаются связи между собой независимых модулей посредством html-шаблонов. Это необходимость возникла для вывода на одной странице информации из различных модулей. Этот механизм позволяет задавать условия, в зависимости от которых одному и тому же модулю могут быть подключены различные шаблоны.

### Инсталляция модулей

Чтобы упростить развертывания функциональных частей web-приложения и обеспечить возможность простого его конфигурирования, необходим механизм установки разработанных модулей в систему. Для этого был разработан специальный формат инсталляционного пакета на основе XML-технологии. Содержимое такого пакета считается одной независимой функциональной единицей, которая при разворачивании в системе сразу же начинает работать.

Для того чтобы прочитать необходимым образом пакет для его дальнейшей инсталляции, разработан класс utilities.system.ParserPackages. Его единственный public метод parse(), придерживаясь заложенной логики, преобразует XML-файл в ассоциативный массив, который методом возвращается в качестве результата. Полученный массив используется для установки всех извлеченных компонентов: модуля и принадлежащих ему сервисов, таблиц, коллекций модельных объектов, нефункциональных связей, менеджеров, action-ов. Далее активные компоненты (другими словами, страницы сайта) добавляются к структуре сайта, затем структура привязывается к стандартным шаблонам. После этих всех автоматически проделанных операций модуль готов к работе.

### Заключение

В ходе работы была описана предметная область и обоснована актуальность проблемы реализации многофункциональных web-приложений. На основании поставленной задачи был изучен опыт в области создания набора средств, ускоряющих процесс разработки, и обозначены ключевые черты framework-системы. Исходя из специфики разработки web-приложений и поставленных задач, были выработаны требования, предъявляемые к framework-системе, и задачи, которые она должна решать. На основе требований разработана логическая архитектура framework-систем. Впоследствии система была успешно апробирована, на ее основе разработано web-приложение для компании Arman. Сайт включает в себя 15 модулей различной функциональности. С ее помощью действительно была повышена эффективность и скорость разработки приложений.

### Литература


1. Олищук А. Теория разработки framework-систем. // PHP Inside. 2004. № 9. С. 5–18.
2. Фаулер М. Архитектура корпоративных программных приложений / Пер. с англ. М.: Издательский дом «Вильямс», 2004. 544 с.
3. Соммервилл И. Инженерия программного обеспечения, 6-е изд.: Пер. с англ. – М.: Издательский дом «Вильямс», 2002. 624 с.

# ИСПОЛЬЗОВАНИЕ WEB-СРЕДСТВ ДЛЯ КОММУНИКАТИВНОЙ ПОДДЕРЖКИ ПРОЦЕССА ПРОЕКТИРОВАНИЯ В РАСПРЕДЕЛЕННОЙ ГРУППЕ

**В.В. Крюков**
**Научный руководитель – к.т.н., доцент Н.Ф. Гусарова**

Обосновывается актуальность проблемы коммуникативной поддержки проектирования в территориально распределенных группах на этапе концептуального проектирования. Предлагается расширить возможности общения внутри проектной группы посредством использования стандартных и вновь разрабатываемых средств обмена и хранения информации, используемых в Web-среде.

## Введение

Актуальность проблемы информационно-коммуникативной поддержки проектирования в территориально распределенных группах определяется тем, что все большее количество предприятий при проведении проектных работ задействует удаленные подразделения или экспертов, территориально находящихся в других городах и странах. В данной ситуации возможно использование традиционных способов удаленного общения, таких как телефон, командировки, видеоконференции или электронная почта. Однако они либо слишком дороги, либо мало эффективны из-за ограничений как по видам коммуникаций, которые они поддерживают, так и по количеству участников, которые одновременно могут общаться между собой.

Обзор [1–3] показал, что большинство существующих САПР ориентировано на проектно-конструкторские работы или ведение архива проектов, а поддержка распределенного проектирования сводится к возможности совместного редактирования группой проектировщиков одного и того же документа. При этом предполагается, что интенсивные коммуникативные процессы, исключительно важные на этапе концептуального проектирования, осуществляются в ходе личного общения. Данная ситуация привела к развитию средств коммуникации на основе web-ресурсов [4], однако существующие решения в данной области предназначаются главным образом для управления проектом и зачастую не имеют достаточной методологической поддержки, особенно для этапа концептуального проектирования. Это делает актуальным разработку организационно-технических систем поддержки концептуального этапа проектирования на основе web -ресурсов.

В работе предлагается расширить возможности общения внутри проектной группы посредством использования стандартных и вновь разрабатываемых средств обмена и хранения информации, используемых в web-среде. В частности, рассматривается практика применения таких средств, как форум, ICQ, электронная почта, а также предложения по разработке дополнительных средств поддержки распределенного проектирования. Приводится концепция объединения системы обмена текстовыми сообщениями в реальном времени с редактором изображений (Flash-PHP чат) и внедрения модуля глоссария в форум проекта. Flash-PHP чат позволяет создавать и обмениваться простыми графическими изображениями и схемами в режиме реального времени без использования внешнего редактора изображений. Интеграция глоссария с форумом позволяет не только хранить данные о специфичных терминах, используемых в проекте, но и явным образом выделять их в текстах сообщений, что облегчит понимание предметной области проекта его участниками.

## Обзор аналогов

Обзор существующих вариантов организации общения в распределенной проектной группе показал, что для этого имеются несколько возможностей, а именно:

- использование возможностей совместной работы, заложенных в традиционных САПР, но они, как правило, ограничиваются совместной работой с чертежами и проектной документацией;
- использование разнообразных технических и информационных систем передачи информации, но они дороги и ограничены в видах передаваемой информации;
- использование не предназначенных для распределенного проектирования программ совместно со средствами передачи данных, однако такой способ неудобен;
- использование средств коммуникации на основе web-ресурсов, но существующее сегодня решения, предназначены главным образом для управления проектом.

## Концепция решения

В качестве дополнительных средств коммуникативной поддержки проектирования предлагается использование Интернет-форума, установленного на web-ресурсе выделенном для ведения проекта. На него дополнительно могут быть установлены: чат, объединенный с графическим редактором – для обсуждения эскизных проектов нового изделия в режиме онлайн; а также глоссарий – для хранения информации об используемых в проекте терминах и помощи в быстром введении в курс дела, новых участников проектной группы. Все инструменты могут быть интегрированы для удобства использования. При этом благодаря модульности конструкции обеспечивается определенная гибкость – не нужные в конкретном проекте инструменты могут быть убраны.

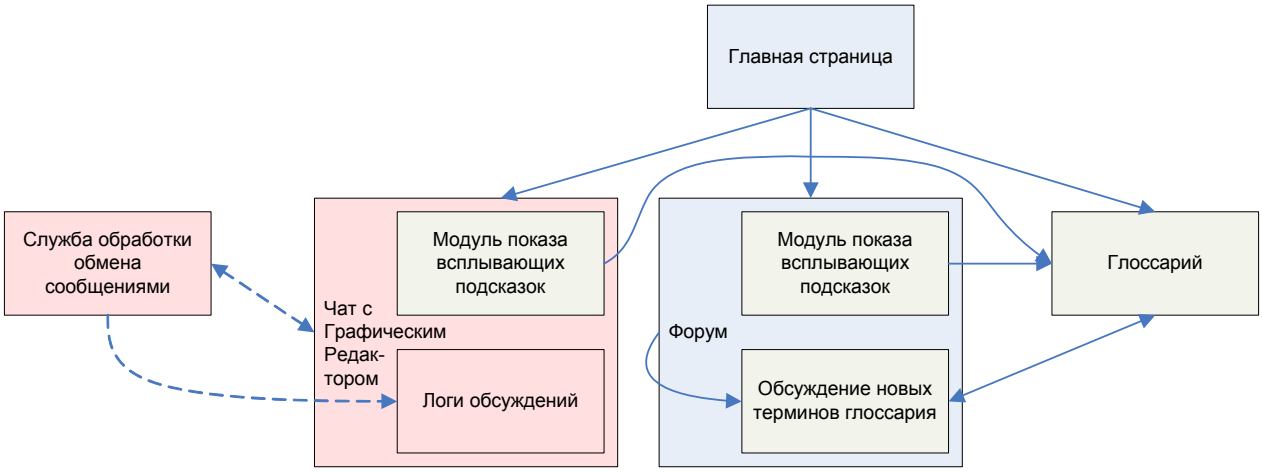



Рис. 1. Структура разрабатываемой системы

Структура системы организационно-технической поддержки концептуального этапа проектирования с использованием разработанного сетевого инструментария представлена на рис. 1. На Интернет-ресурс проекта устанавливается форум, на котором участники ведут обсуждение вопросов, не требующих интенсивной коммуникативной деятельности. Также форум выполняет функцию автоматического стенографирования, поскольку сообщения всех пользователей сохраняются и доступны для дальнейшего анализа другими участниками, в том числе присоединившимися к проекту позднее. Для них же предусмотрен модуль глоссария, который в базовом варианте встраивается в форум, однако также может быть внедрен и на другие страницы ресурса проекта и в чат. Он обеспечивает подсветку и вывод определений всех терминов, используемых участниками при обсуждении и имеющимися в его словаре. Для согласования новых терминов предусмотрена возможность их обсуждения в специальном разделе форума с последующим занесением их в глоссарий. Также имеется возможность просмотра всех терминов глоссария. Наконец, для обсуждений требующих интенсивной коммуникативной деятельности, предусмотрен чат с графическим редактором. При реализации его на связке flash и php, клиентская часть

встраивается в любую страницу ресурса или форума, а серверная устанавливается на сервер, в качестве службы. Возможно сохранение логов для их дальнейшего анализа.


Рис. 2. Схема движения информационных потоков при использовании составлении документа, с использованием разрабатываемой системы

Схема движения информационных потоков при решении задачи, например, составления документа с предпроектными предложениями, представлена на рис. 2. Вначале инициатор или инициативная группа, подают запрос; затем, группа разработчиков начинает выдвигать и обсуждать идеи с использованием предложенных средств, в результате чего формируется документ, который в случае утверждения инициатором дискуссии попадает в хранилище документации или передается на последующие этапы проектирования.

### Объединение чата с графическим редактором

Объединение в единый интерфейс системы обмена текстовыми сообщениями и графического редактора предоставит проектировщикам новые возможности при общении в режиме реального времени в рамках группы. В отличие от традиционного чата или ICQ, они смогут обмениваться не только текстовыми сообщениями, но и эскизами или структурными схемами будущих изделий, без необходимости тратить время на пересылку файлов по электронной почте и использования вспомогательных программ (рис. 3). Использование единого рабочего поля позволит нескольким участникам работать с одним и тем же изображением, как в обычных САПР, поддерживающих совместное проектирование. При реализации данного решения на связке flash и php для работы приложения достаточно обычного браузера без необходимости устанавливать специализированные пакеты программ, что немаловажно, если кто-то из участников выходит в сеть с общественного компьютера.

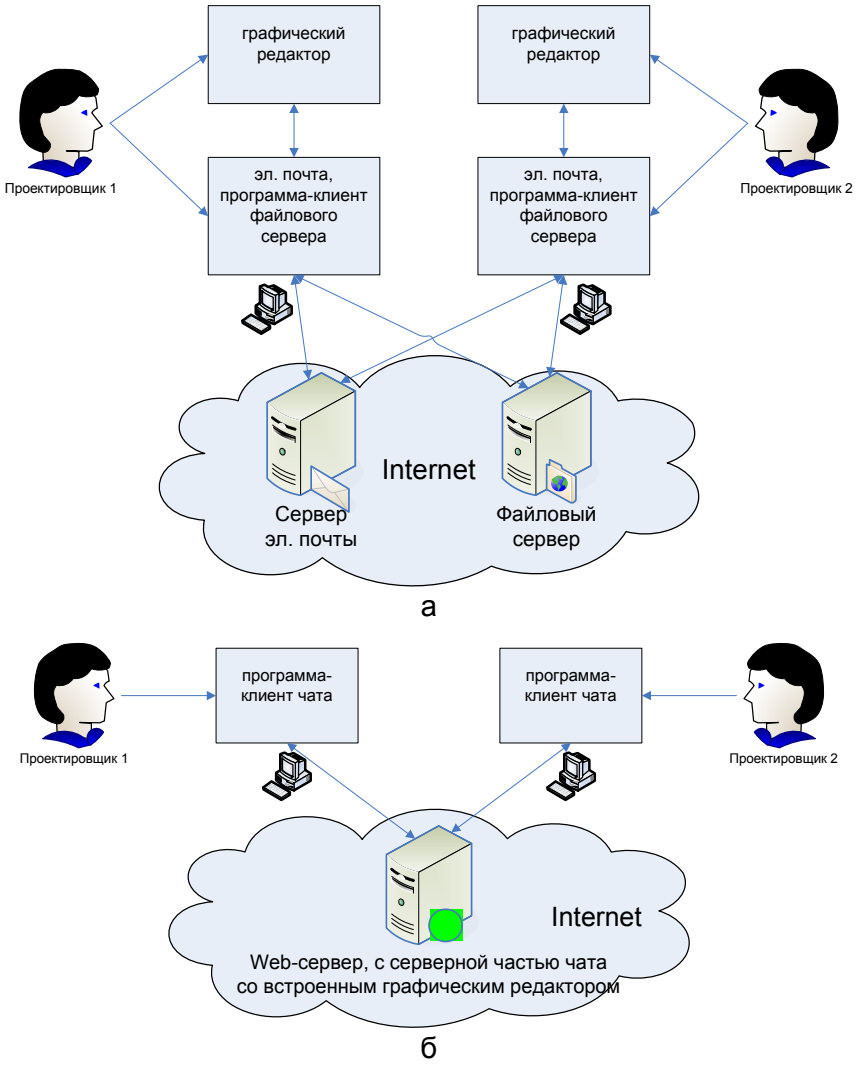



Рис. 3. Схема обмена информацией в распределенной группе проектирования: а – традиционная; б – при использовании чата со встроенным графическим редактором

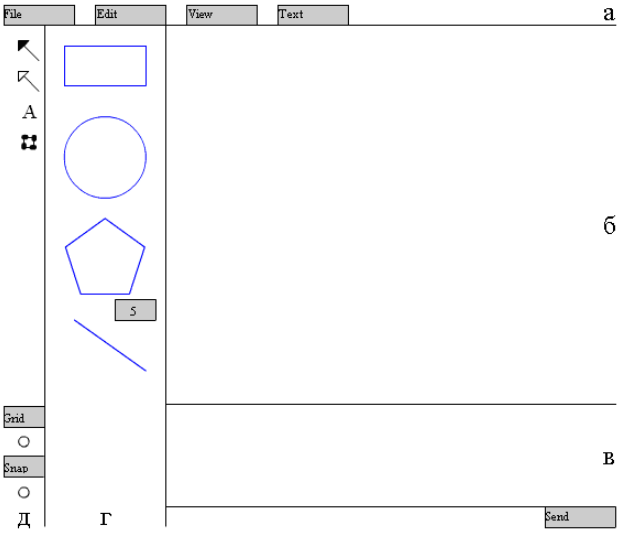



Рис. 4. Примерный вид интерфейса приложения, справа на лево: а – меню для создания и доступа к проектам, подключения внешних модулей и редактирования параметров приложения; б – рабочее поле; в – классический текстовый чат; г – набор графических примитивов; д – линейка инструментов

**Использование глоссария для поддержки процесса проектирования**

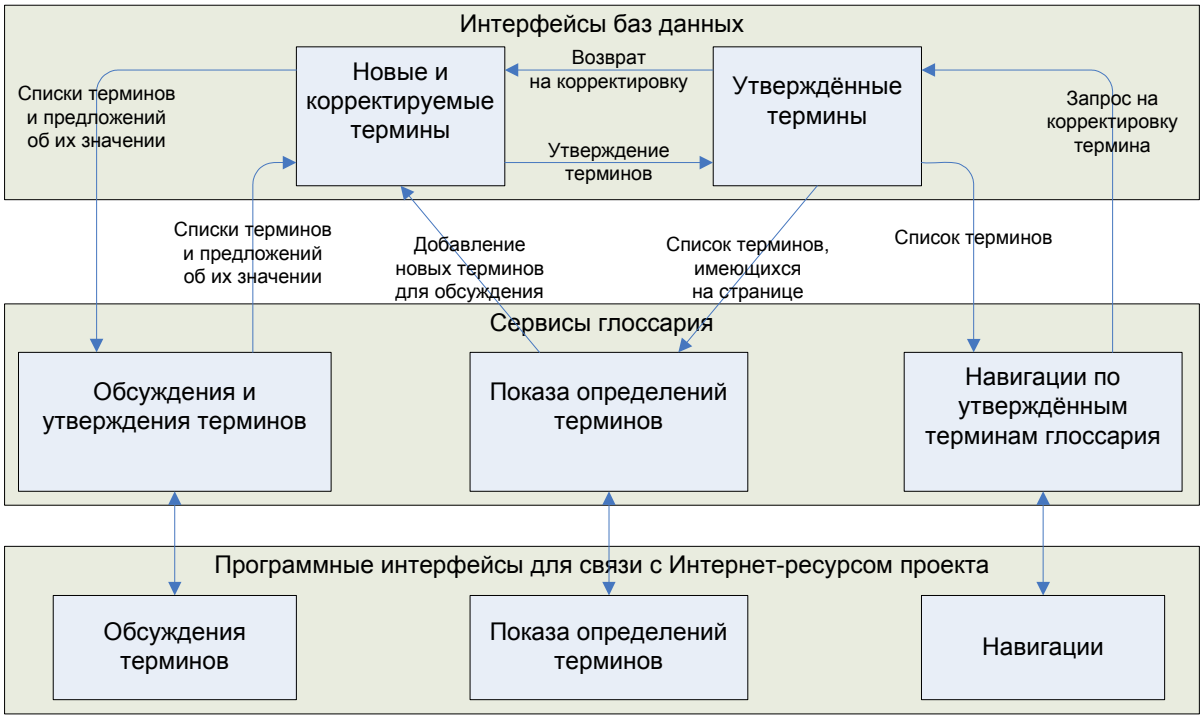



Рис. 5. Архитектура модуля глоссария

Согласно стандартам PMI [5], любой проект начинается с обсуждения и однозначно трактуемого определения всех терминов предметной области, которое фиксируется в разделе «глоссарий». Для глоссария поддерживающего распределенную разработку проекта, были выделены следующие требования:

- описания терминов должны быть поняты и приняты всеми участниками и документально зафиксированы;
- глоссарий должен корректироваться и дополняться в течение всего жизненного цикла проекта;
- глоссарий должен интегрироваться в ресурс, используемый для распределенного проектирования.

Этим требованиям удовлетворяет модуль глоссария, архитектура которого представлена на рис. 5. Использование подобного глоссария позволит хранить определения всех терминов используемых в проекте, а его интеграция с ресурсом проекта, позволит явно выделять эти термины в текстах сообщений, что облегчит понимание предметной области проекта его участниками.

**Заключение**

Данные средства использовались при разработке образовательного ресурса СПбГУ ИТМО и портала о машино- и приборостроении «ПервоМаш».

Благодаря их использованию удалось повысить качество и наглядность концептуального этапа проектирования, эффективно и практически без материальных затрат, осуществлять процесс проектирования когда один из разработчиков находился в другой стране.

Данные результаты подтверждают эффективность применимости данных средств при решении задач концептуального проектирования и поддержки жизненного цикла различных информационных систем, в том числе ориентированных на приборостроение.

В качестве направлений дальнейших исследований, наиболее перспективными выглядят углубление интеграции модулей в единую систему, проработка вопросов обеспечения безопасности и разработка методики организационно-технической поддержки концептуального этапа проектирования.

## Литература


1. ППП CATIA компании Dessault Systemes. http://www.catia.ru/
2. ППП Autodesk Inventor. http://www.inventor.ru/
3. Башмаков А.И., Башмаков И.А. Интеллектуальные информационные технологии. М.: МГТУ, 2005. 304 с.
4. Техническое описание EMC Documentum eRoom. http://www.documentum.ru/pdf/WhitePapers/WP_eRoom_proj_mgt_rus.pdf
5. Михеев В.Н., Товб А.С. Международные и национальные стандарты по управлению проектами, менеджменту проектов и профессиональной компетентности менеджеров проектов. Труды 2-й Всероссийской практической конференции «Стандарты в проектах современных информационных систем», М., 2002. С. 33–37.

# МЕТОДЫ И АЛГОРИТМЫ РАЗРАБОТКИ АВТОМАТИЗИРОВАННОГО РАБОЧЕГО МЕСТА ПРОЕКТИРОВЩИКА ТЕХНОЛОГИЧЕСКИХ СИСТЕМ

**А.С. Федотов**
**Научный руководитель – д.т.н., профессор Ю.А. Гатчин**

В статье рассказывается о методах и алгоритмах позволяющих разработать автоматизированное рабочее место проектировщика технологических систем. Приводится комплекс необходимых программных продуктов, их анализ и качество работы.

## Введение

Современные технологические системы отличает большое многообразие компонентов и элементов, что увеличивает время проектирования, а как следствие – и их стоимость. Темпы проектирования таких систем являются основным сдерживающим фактором их разработки и внедрения. Решение этой задачи приводит к необходимости создания системы автоматизированного проектирования.

Автоматизирование проектирования технологических систем предполагает разработку специализированных технических средств, обеспечивающих ввод и вывод информации, разработку автоматизированных рабочих мест проектировщика, содержащих комплексы оборудования, а также разработку математического обеспечения: методов, алгоритмов и программ.

Традиционные методы проектирования технологических систем имеют низкую степень формализации и в основном ориентированы на принятие решений проектировщиком, они включают большой объем вычислительных операций, характеризуются низкой точностью, так как очень часто автоматизированное рабочее место проектировщика организовано не на должном уровне. Все это ограничивает их применение. В связи с этим совершенствование традиционных методов, разработка новых методов зачастую неэффективно.

В последние годы появилось много новых методов и программ для проектирования технологических систем. Эти методы отличаются от классических более высокой сложностью, они формализованы, их исполнение связано с большим объемом вычислений, что делает полезным при решении практических задач наличие в них библиотеки стандартных подпрограмм. Однако даже наличие таких библиотек требует от проектировщика значительных усилий в программировании для решения конкретной задачи. Высококачественная, хорошо отлаженная программа, написанная программистом высокой квалификации специально для некоторого проекта, наиболее оптимальна. Развитие технологических систем на таком высоком уровне требует нового подхода к методам и алгоритмам разработки автоматизированных рабочих мест проектировщика, способствует использования последних новинок в области программных продуктов для разработки автоматизированных рабочих мест.

## Структура и назначение автоматизированного рабочего места

Автоматизированное рабочее место (АРМ) (рис. 1) – индивидуальный комплекс аппаратных и программных средств, предназначенный для автоматизации профессионального труда специалиста – картографа, проектировщика электронных схем, оператора системы дальнего радиолокационного обнаружения и пр. Обычно в АРМ входит персональный компьютер или рабочая станция с графическим или текстовым дисплеем, графопостроитель и другие периферийные устройства. АРМ работает в составе локальной или территориальной сети или в автономном режиме [1].


Рис. 1. Схема автоматизированного рабочего места

В настоящее время всестороннее развитие технологии приводят к необходимости выживания промышленных предприятий в новых экономических условиях, осуществляя глубокую конверсию основного производства. Возникают задачи проектирования все более сложных технических объектов в сжатые сроки, требующие специфического оборудования, новых технологий и программных продуктов, а также увеличения численности проектировщиков. Удовлетворить противоречивые требования с помощью простого увеличения численности проектировщиков нельзя, так как возможность параллельного проведения проектных работ ограничена, а численность инженерно-технических работников в проектных организациях не может быть сколько-нибудь заметно увеличена [2]. Выходом из этого положения является широкое применение методов развития и усовершенствования АРМ. Все это приводит нас к автоматизации системы проектирования.

Автоматизированная система проектирования технологий представляет собой одну из составных частей АРМ технолога-проектировщика, главными элементами которой являются, с одной стороны, проектировщик, а с другой стороны – система автоматизации проектирования, т.е. система, предназначенная для совершенствования процесса проектирования, основанная на взаимодействие технического, алгоритмического, программного и информационного обеспечения.

Построение автоматизированной системы проектирования на основе современных информационных технологий и технологий программирования даст возможность расширения программного обеспечения за счет соответствия стандартам построения открытых систем, ведения единой информационной модели для решения технологических задач, интеграции решения задач разного направления, автоматического создания и адаптации математической модели задачи и исходных данных под реальные ситуации, использования графического интерфейса пользователя, упрощающего взаимодействие пользователя с ЭВМ. Особый интерес в настоящее время вызывает подход, при котором проектировщик технологических систем в процессе диалога с системой осуществляет творческое конструирование и выбирает наилучшее проектное решение. Для этого ему необходимо создать АРМ, полностью удовлетворяющее его критериям.

АРМ должно отвечать следующим требованиям:

- своевременное удовлетворение информационной и вычислительной потребности специалиста;
- минимальное время ответа на запросы пользователя;
- адаптация к уровню подготовки пользователя и его профессиональным запросам;
- простота освоения приемов работы на АРМ и легкость общения, надежность и простота обслуживания;
- терпимость по отношению к пользователю;

- возможность быстрого обучения пользователя;
- возможность работы в составе вычислительной сети.

**Новые возможности в свете последних достижений**

Один из методов облегчения работы проектировщика связан с созданием различных автоматизированных баз данных. Специалистам часто приходится работать с большими объемами данных, чтобы найти требуемые сведения для подготовки различных документов. Для облегчения такого рода работ были созданы системы управления базами данных (СУБД: DBASE, RBASE, ORACLE и др.). СУБД позволяют хранить большие объемы информации, и, что самое главное, быстро находить нужные данные. Так, например, при работе с картотекой постоянно нужно перерывать большие архивы данных для поиска нужной информации, особенно если карточки отсортированы не по нужному признаку. СУБД справится с этой задачей за считанные секунды [2].

Методом разработки АРМ проектировщика технологических систем можно считать метод использования различных программных комплексов. Программный комплекс LCAD (от Layout CAD – расстановка оборудования с помощью компьютера) предназначен для создания АРМ проектировщика, осуществляющего технологическое проектирование новых производственных помещений (рис. 2), а также технологическую реорганизацию существующего производства. Комплекс может быть также использован для получения различной справочной информации по установленному на производстве и введенному в базу данных системы оборудованию.

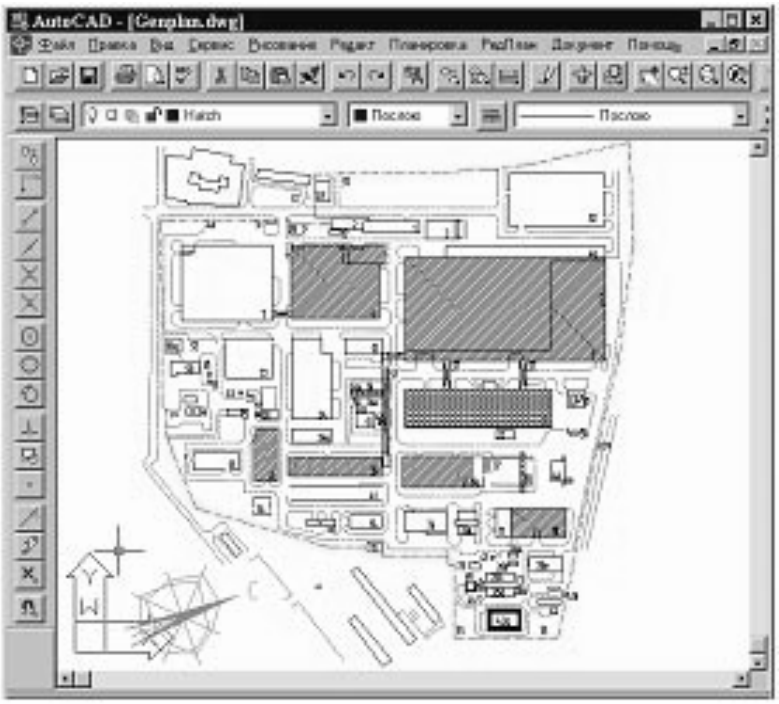



Рис. 2. Проектирование новых производственных помещений

Программный комплекс LCAD позволяет автоматизировать процесс формирования:

- строительной подосновы (планов этажей зданий) по одноэтажным и многоэтажным промышленным помещениям для последующего размещения технологического оборудования; а также административно-бытовым зданиям;
- графической и текстовой документации по технологической планировке производственных помещений.

LCAD обеспечивает создание и ведение базы данных (БД), содержащей массивы текстовой и графической информации. Структура массивов БД позволяет загружать и использовать при проектировании следующие виды информации:

- характеристики оборудования (наименование и модель, габариты, масса, установленная мощность электродвигателя и некоторая дополнительная информация), с обеспечением поиска и выбора информации по классам и группам оборудования;
- дополнительная графическая информация по оборудованию: размеры, установочные планы, планы опор, точки подключения электропитания, воздуха и т.п.;
- темплеты («габаритки», «фишки») оборудования;
- спецификации по установленному оборудованию;
- принятые условные графические обозначения для нанесения на планировки;
- структура производства (промышленная площадка – производственный корпус – цех – участок);
- генплан предприятия (для обеспечения быстрого выхода на нужную планировку производственных корпусов, цехов, участков);
- любая информация по цехам и участкам предприятия (виды и размеры площадей и т.д.);
- справочные данные по нормам и требованиям к размещению оборудования.

LCAD предполагает создание и хранение в БД технологических планировок на строительной подоснове производственного корпуса (здания) в целом. Спецификация установленного оборудования (рис. 3) создается и хранится в БД в целом по предприятию. Оформление и вывод на печать графической (чертежи планировок) и текстовой (спецификации оборудования) документации может производиться как в целом по производственным корпусам, так и по отдельным цехам и участкам, запрашиваемым в БД.

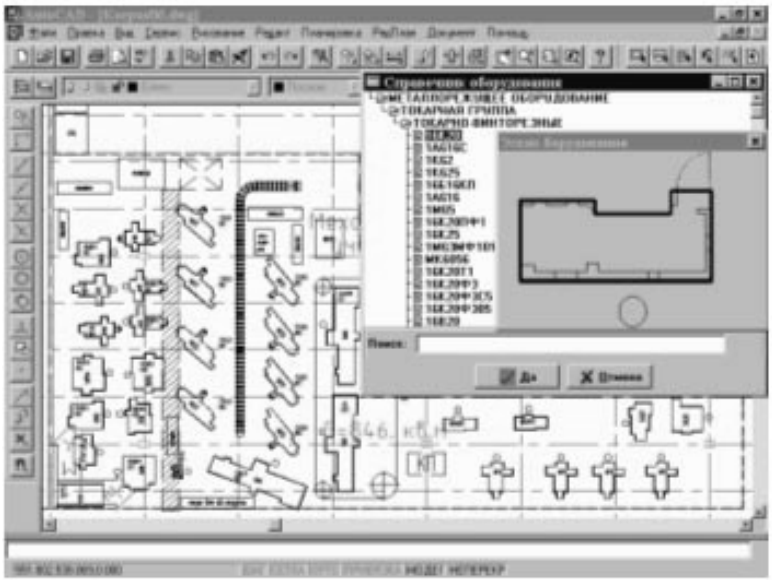


Рис. 3. Спецификация установленного оборудования

LCAD использует и расширяет возможности пакета AutoCAD фирмы Autodesk за счет наличия дополнительного набора специальных приложений, обеспечивающих основные функции проектирования технологических планировок цехов и участков предприятии [3]. Комплекс можно использовать в технологических подразделениях и технических отделах как крупных предприятий, так и небольших производственных организаций, применяющих АРМ технологов-проектировщиков на базе персональных компьютеров.

Великолепные достижения современной информатики, большое количество и значительный ассортимент программных продуктов позволяют строить процесс проектирования на новом, совсем недавно недоступном, уровне. Обобщая доступные знания

о современных достижений, можно попробовать виртуально синтезировать АРМ проектировщика. Рассмотрим АРМ проектировщика изделий электронной техники. Основной метод создания такого АРМ, как и многих других, основан на внедрении последних программных продуктов. Необходимыми составными частями предлагаемого АРМ являются графический, топологический и текстовый редакторы, а также пакет программ схемотехнического моделирования.

Сердцем виртуального АРМ выбираем графический редактор, а именно AutoCAD200X фирмы AutoDesk. Дружеское и квалифицированное присутствие его создателей ощущается в процессе всей работы. Действительно, трудно найти такой режим работы конструктора, который бы не предусмотрели специалисты AutoDesk. Интерфейсное окно одного из замечательных продуктов AutoDesk, а именно – AutoDesk Mechanical Desktop 2004, с примером трехмерного отображения сборочного чертежа платы, приведено на рис. 4.

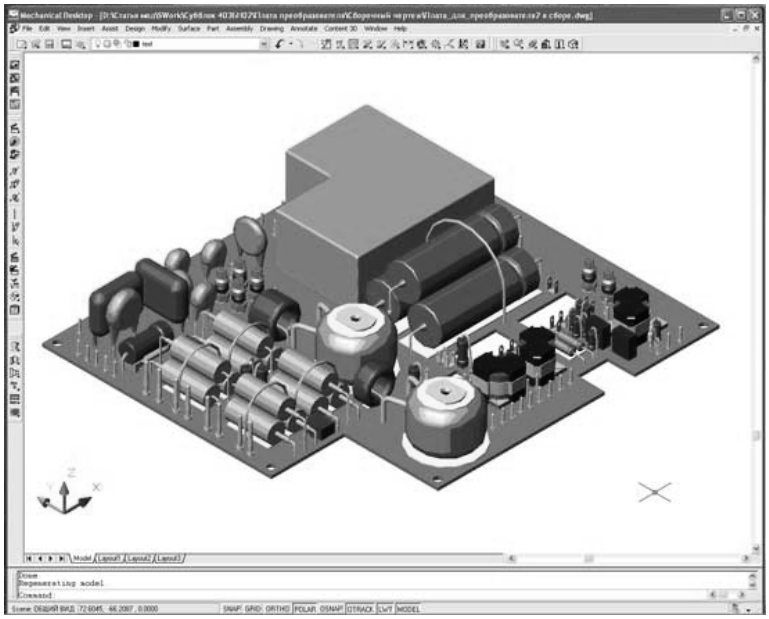


Рис. 4. Интерфейсное окно AutoDesk Mechanical Desktop 2004 с примером трехмерного отображения сборочного чертежа платы

Руками нашего создания будут являться топологический и текстовый редакторы. Правой рукой назначаем топологический редактор, а конкретно PCAD200X. Основным и несомненным достоинством в его работе является наличие функции автотрассировки (особенно при наличии программы SPECCTRA). Действительно, получение готовой топологии платы на основе схемы электрической принципиальной и грамотно составленного задания является значительным шагом в автоматизации и, соответственно, облегчении работы тополога, тем более в таком рутинном сегменте разработки. Левой рукой у нас будет текстовый редактор. Несомненным лидером в этой номинации является Microsoft Word, его и возьмем в помощники. Ну а головой, конечно же, является пакет программ схемотехнического моделирования. С середины 90-х годов прошлого века автор успешно эксплуатировал пакет Design Lab, добиваясь значительно большей производительности, чем в современных OrCAD 9.X, OrCAD 10, причем на менее мощных компьютерах. Приходится только мириться с некоторыми «дикими» зигзагами развития рынка. Безусловно, достоинством семейства OrCAD является наличие программы схемотехнического и функционального моделирования Capture OrCAD [4].

Описанные выше функциональные возможности отсутствуют в каждой из составных частей нашего АРМ, однако они присутствуют в пакетах программ некоторых машинных станций и платформ типа UNIX, Hewlett-Packard и других. И хотя их стоимость и стоимость предлагаемого комплекса разнятся многократно, ставится задача

осуществления таких функциональных возможностей, которые бы не только превосходили аналоги, но и переводили бы систему на качественно новую ступень. За теоретическую основу примем статью [5]. Действительно, производительность современных персональных компьютеров позволяют наделить уже не электронные устройства, а рассматриваемый программный комплекс абсолютно новыми свойствами. Определим появление комплекса новых свойств как «интеллектуализация программного продукта».

С целью построения алгоритмов действия промоделируем процесс разработки, осуществляемый человеком, и попробуем перенести выявленные закономерности в деятельность создаваемого продукта.

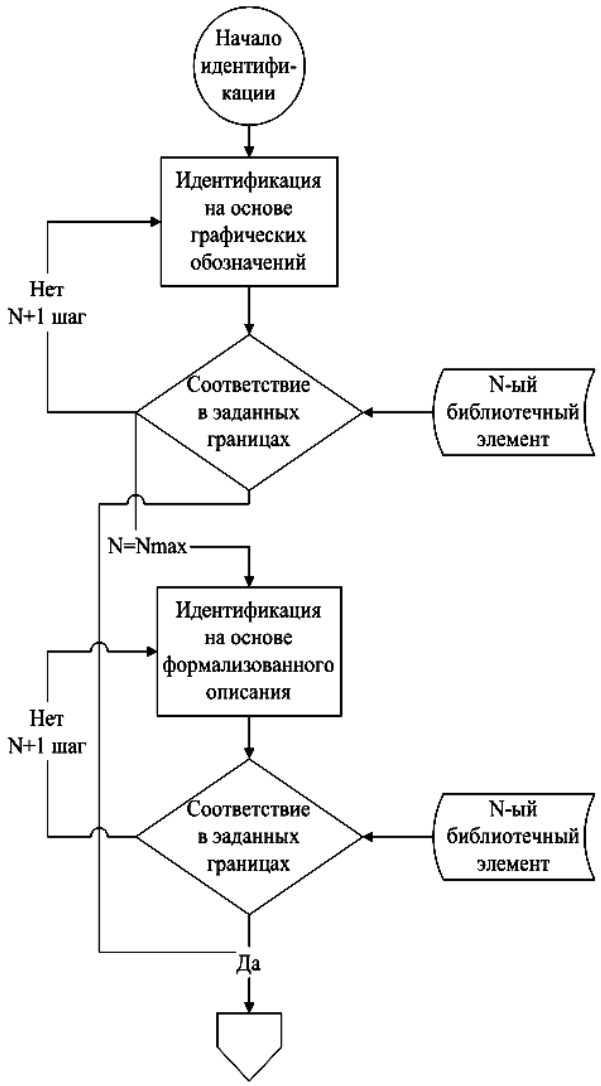



Рис. 5. Алгоритм процесса идентификации

Начнем с этапа получения задания. Получив задание, человек ищет в памяти (своей базе данных) аналоги заданию, с целью его возможного выполнения путем модернизации. Рассмотрим программное осуществление данного этапа. Дополнив графическое обозначение кратким формализованным описанием, мы получим элемент идентификации для АРМ. Этот элемент будет использоваться не только при вводе задания, но и для краткого формализованного описания разработанных продуктов. Храниться он будет в отдельной библиотеке, а задействован будет для идентификации разработок. Механизм идентификации на начальном этапе будет несовершенен из-за недостаточной формализации задействованных в процессе данных. Но попробуем обойти эти трудности, используя информационную избыточность: если не удается идентифицировать разработ-

ку с помощью системы графических обозначений, то используется система формализованного описания. Алгоритм предлагаемого процесса проиллюстрирован рис. 5.

После процесса идентификации наступает этап конкретного сопоставления полученных в задании данных и параметров идентифицированной разработки. С этой целью мы переходим от функциональных моделей к иным моделям. Эти модели могут предоставить нам интересующие нас данные. Вот тут мы сразу же вспоминаем, что рабочее место у нас автоматизированное, а не автоматическое. Дело в том, что огромный объем используемой информации различного вида, и сложнейшие алгоритмы действий не позволяют автоматизировать этот процесс на современном этапе. Да и, в конце концов, разработчик должен продемонстрировать творческое начало. Но и в этом случае современные программные продукты могут оказать неоценимую помощь. Одним из таких инструментов является директива вариации параметров программы Spice. Но целью статьи является не обучение пользователей, а стремление показать возможность создания высококачественного АРМ [5].

## Заключение

Если обратить внимание на достижения данного направления, то подавляющее место в нем занимают зарубежные продукты, что весьма обидно.

Предлагаемое АРМ окажется полезным системотехникам, схемотехникам, конструкторам, топологам и технологам. Пользователю не обязательно устанавливать четыре–пять редакторов. Можно обойтись минимально количеством, необходимым для работы. Но сами закладываемые в предлагаемый программный продукт принципы не только адекватны возникающим в процессе разработки задачам, но и переводят процесс разработки на более высокий уровень. Внедрение проекта позволит не просто значительно автоматизировать процесс разработки и создавать интегрированные библиотеки разработок, но и создать рынок разработок, не имеющий аналогов.

Анализируя сущность АРМ, специалисты определяют их чаще всего как профессионально-ориентированные малые вычислительные системы, расположенные непосредственно на рабочих местах специалистов и предназначенные для автоматизации их работ. Для каждого объекта управления нужно предусмотреть АРМ, соответствующие их функциональному назначению. Однако принципы создания АРМ должны быть общими: системность, гибкость, устойчивость, эффективность.

## Литература


1. Словарь по естественным наукам. http://slovari.yandex.ru/
2. Козлова Е.В., Когутенко В.А. Модифицированный метод структурного распараллеливания В.А. Костенко для линейных и разветвляющихся участков схемы технологического процесса сборочного производства.
3. www.cad.ru
4. Силкин В. Трехмерное отображение в электронике - варианты использования и возможные направления развития. // Компоненты и технологии. 2005. №5. С. 26–28.
5. Силкин В. «Интеллектуализация» электронных устройств. // Компоненты и технологии. 2005. №3. С. 37–39.

# ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ СОВРЕМЕННЫХ ВСТРАИВАЕМЫХ ЭЛЕКТРОННО-ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ И РАЗРАБОТКА ПЛАТ ДЛЯ ПРОТОТИПИРОВАНИЯ

**П.А. Косенков, А.О. Терентьев**
**Научный руководитель – д.т.н., профессор Ю.А. Гатчин**

Рассматриваются особенности проектирования встраиваемых микропроцессорных систем, разработаны варианты плат для прототипирования, позволяющие ускорить процесс проектирования встраиваемых устройств.

## Введение

Основной особенностью современных встраиваемых ЭВС является то, что они приобретают характеристики, свойственные ранее только настольным или большим ЭВС. Эти характеристики, такие как использование файловых систем, средств сетевых коммуникаций и графических пользовательские интерфейсов, вызывают серьезный рост системных требований. Вместе с тем встроенные ЭВС должны отвечать и дополнительным требованиям, связанным с их «встраиваемой» спецификой. Это малые габариты, низкое энергопотребление, вибростойкость, широкий диапазон рабочих температур, и другие. Требования, предъявляемые к данным системам, являются более жесткими по сравнению с требованиями к настольным или большим ЭВС. Это вызывает необходимость разработки таких систем с использованием более совершенных, нежели ранее, средств и методов.

## Подходы к разработке встроенных систем и их особенности

Существует несколько подходов, позволяющих сэкономить ресурсы и время на разработку системы. Современная элементная база и наработанное программное обеспечение позволяет создавать интегрируемые решения в короткие сроки. Из распространенных на сегодняшний момент систем можно выявить 3 группы по применяемым процессорным архитектурам:

1. применение центрального процессора на базе RISC, MIPS или другой «жесткой» архитектуры;
2. применение специализированных DSP процессоров, предназначенных для обработки большого количества потоковой информации;
3. применение ПЛИС, внутренняя архитектура которых создается на основании требований встроенной системы.

**Центральный процессор общего назначения.** Данная архитектура применяется там, где требуется обработать зачастую уже логические сигналы. Существует множество направлений и подходов в данном сегменте встроенных систем:

1. применение универсальных мезонинных компьютеров, построенных на уже устоявшихся архитектурах, таких как x86;
2. применение RISC архитектуры, в частности, построенной по технологии ARM (Advanced RISC Machines);
3. применение MIPS (Microprocessor without Interlocked Pipeline Stages – «микропроцессор без блокировок в конвейере») архитектуры с относительно длинным конвеером.

**Сигнальный процессор.** Особенности этой архитектуры заключается в структуре ядра, адаптированного для обработки сигналов в цифровом виде. Для нее характерны:

- быстрое выполнение операций цифровой обработки сигналов, например, операция «умножение с накоплением» исполняется за один такт;

- «бесплатные» по времени циклы с заранее известной длиной;
- довольно большой объем встроенной памяти, из которой может осуществляться выборка нескольких машинных слов одновременно;
- детерминированная работа с известными временами выполнения команд, что позволяет выполнять планирование работы в реальном времени;
- довольно большая длина конвейера, в результате чего незапланированные условные переходы занимают относительно много времени;
- экзотический набор регистров и инструкций, часто неудобный для компиляторов;
- ограниченный, по сравнению с микроконтроллерами, набор периферийных устройств – впрочем, существуют «переходные» чипы, сочетающие в себе свойства DSP и широкую периферию микроконтроллеров, и даже отдельное RISC ядро, например, серия TMS320C24xx компании Texas Instruments [1];
- цифровые сигнальные процессоры обычно потребляют существенно меньше мощности, чем эквивалентные по производительности процессоры общего назначения.

**Программируемые логические интегральные схемы.** В отличие от обычных цифровых микросхем, логика работы ПЛИС не определяется при изготовлении, а задается посредством программирования. Для программирования используются языки Verilog [2], VHDL [3]. Основные современные типы ПЛИС:

- CPLD [4] (complex programmable logic) содержат относительно крупные программируемые логические блоки – макроячейки (macrocells), соединенные с внешними выводами и внутренними шинами. Функциональность CPLD кодируется в энергонезависимой памяти, поэтому нет необходимости их перепрограммировать при включении;
- FPGA (field-programmable gate array) содержат логические элементы и блоки коммутации. FPGA обычно имеют больше логических элементов и более гибкую архитектуру, чем CPLD. Программа для FPGA хранится в распределенной оперативной памяти микросхемы, поэтому требуется начальный загрузчик.

Вышеперечисленные подходы обладают нечеткими границами, и зачастую возможны комбинации. Например, все больше присутствуют на рынке высокоинтегрированные системы, совмещающие в себе несколько подходов – например, линейка 2-ядерных процессоров от Texas Instruments [1], где на одном кристалле совмещены DSP и ARM архитектура.

В современном проектировании намечается тенденция к использованию уже готовых открытых программных разработок на основе Unix-совместимых операционных систем. Такой подход позволяет сэкономить на этапе подготовки встроенного программного обеспечения и сократить производственный цикл на данном этапе создания встроенной системы. А привлечение сообщества программистов после выпуска устройства позволяет поддерживать актуальность разработки неопределенно долгое время благодаря личной заинтересованности каждого в совершенствование конечного продукта.

Сегодня большую популярность приобретают встраиваемые программные среды с открытым исходным кодом. Такой подход удобен не только разработчикам, но и производителям компонентов, так как создается база готовых программных решений и сообщество разработчиков, применяющих компоненты на производстве. Ярким примером может служить аппаратно-программный комплекс DaVinci от Texas Instruments [1, 5].

Под общим именем объединены уже выпускающиеся микропроцессоры и специально разработанное программное обеспечение, включающее в себя как инструменты отладки с обширной базой технологических решений, так и открытую программную архитектуру на основе Unix-подобных систем, комбинации которых позволяют создавать унифицированные видеовоспроизводящие и записывающие устройства, встраи-

ваемы системы контроля доступа и обработки аналоговых сигналов в реальном времени. В частности, авторы данной статьи ведут разработки в области разработки программно-аппаратного комплекса автоматизированного проектирования таких встроенных электронно-вычислительных систем.

## Разработанные варианты плат прототипирования для программно-аппаратного комплекса автоматизированного проектирования встраиваемых устройств

Проектирование современных встроенных систем включает в себя несколько этапов:

- определение требований системы;
- функциональное описание;
- выбор процессорной системы;
- разработка аппаратного обеспечения;
- разработка программного обеспечения;
- интеграция.

На этапе разработки аппаратного и программного обеспечения проекта используются прототипы плат, позволяющие проверить ее работоспособность и изучить возможности проектируемой системы.

В ходе проделанной одним из авторов (Косенковым П.А.) работы в открытом проекте [6] был разработан набор плат для прототипирования на базе микропроцессора архитектуры ARM (at91rm9200) и ПЛИС Altera Cyclone EP1C3T144.

**Плата на базе микропроцессора архитектуры ARM (at91rm9200).** Плата построена на базе процессора ARM920T с производительностью 200 млн. инструкций в секунду и содержит 16 кбайт кэш-памяти инструкций и 16 кбайт кэш-памяти данных, 16 кбайт статического ОЗУ, 128 кбайт ПЗУ, интерфейс внешней шины EBI с поддержкой синхронного динамического ОЗУ (SDRAM), контроллеры флэш-памяти с пакетным режимом передачи данных и статической памяти, главный USB-порт, USB-порт устройства, модуль Ethernet 10/100 Base T MAC, контроллер управления энергопотреблением, часы реального времени, системный таймер, синхронный последовательный контроллер, 6-канальный таймер-счетчик, 2-проводной интерфейс, последовательный периферийный интерфейс (SPI), интерфейсы мультимедиа-карт и контроллер параллельного ввода-вывода.

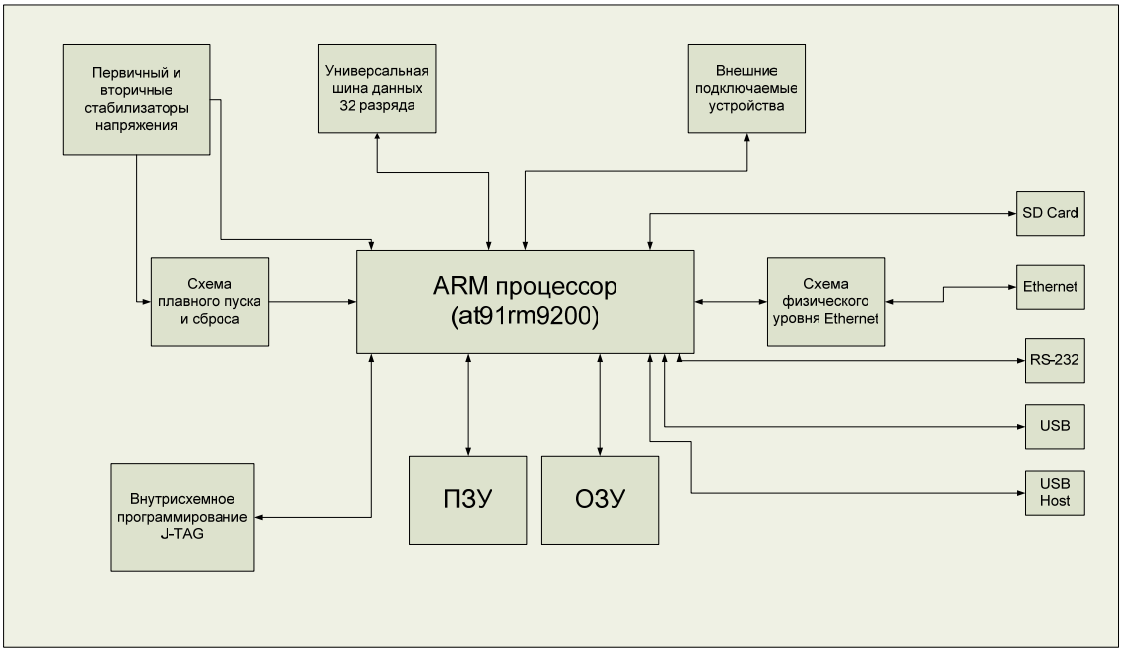



Рис. 1. Функциональная схема прототипа с ARM архитектурой

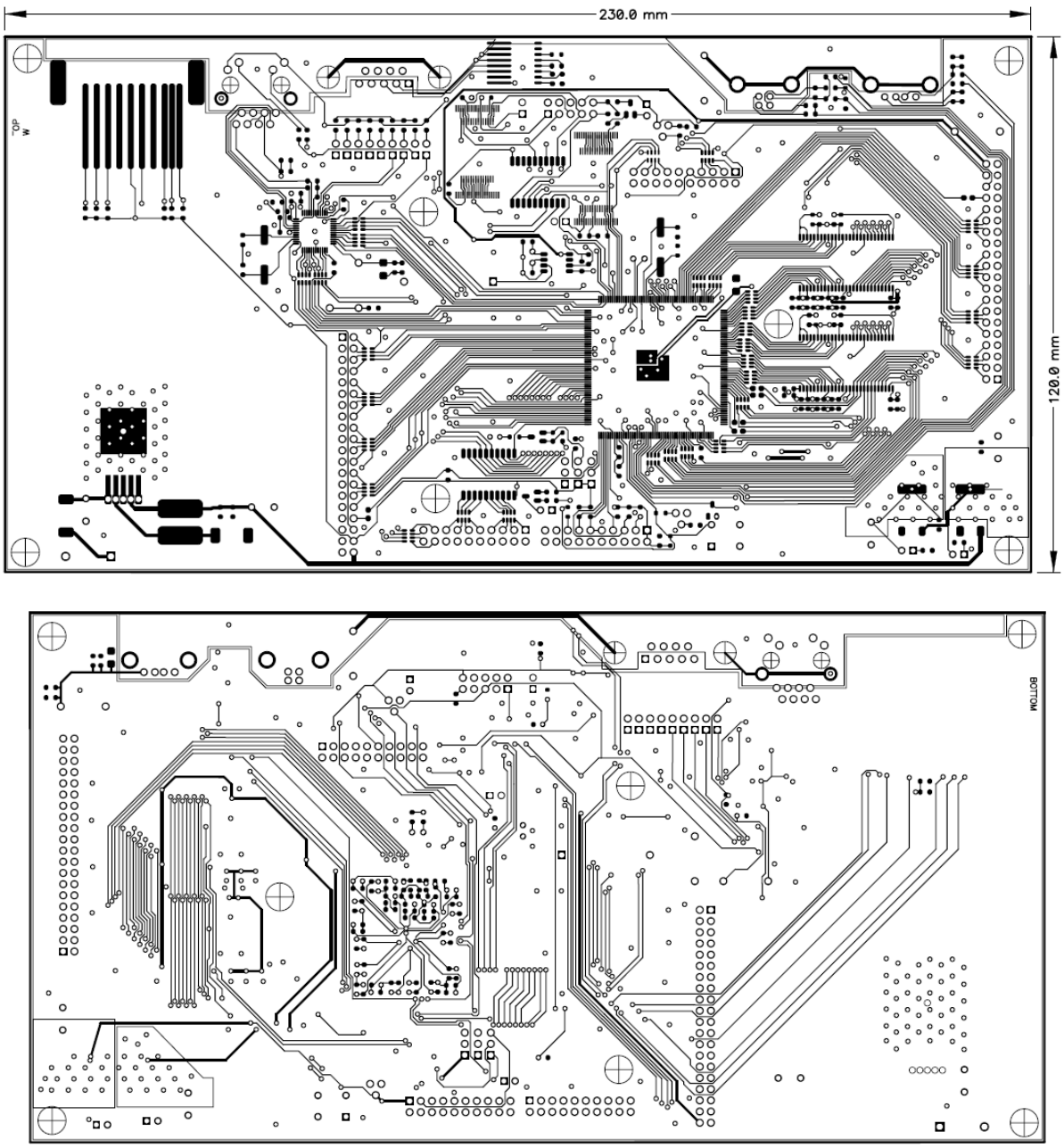



Рис. 2. Топология печатной платы прототипа с ARM архитектурой

Функциональная схема платы приведена на рис. 1. В работе реализована основная часть доступного функционала данного микропроцессора. Плата обладает возможностью установки до 128 мБ ОЗУ, до 32 Мб ППЗУ (Flash). Доступен широкий профиль периферии:

- RS-232 для отладки и организации терминального доступа;
- Ethernet 10/100 Base T на базе микросхемы KS8721BL;
- полноскоростной порт USB 2.0 (ведущий и ведомый);
- возможность работать с флеш-картами SD/MMC;
- JTAG интерфейс для отладки встраиваемого программного обеспечения.

Помимо стандартных интерфейсов, плата обладает 2×16-битных шин ввода-вывода данных и дополнительные 20 универсальных линий для возможности подключения дополнительных модулей. Топология печатной платы представлен на рис. 2. Трассировка платы выполнена в двухслойном исполнении в программном комплексе P-Cad 2002 преимущественно в ручном режиме. По причине достаточности для отладки

уже доступного функционала, не реализованными остались такие возможности, как использование CF карты памяти, IDE интерфейс, интерфейс для подключения графического дисплея, IrDA интерфейс, Контроль параметров питающих цепей. Эти возможности планируется реализовать в следующих вариантах платы.

Данная плата допускает установку встраиваемой версии Unix-совместимой операционной системы, в частности uClinux. Такой подход позволяет получить систему обработки сигналов в реальном времени и получить комплекс программных и аппаратных инструментов для обработки, хранения и передачи данных.

**Плата на базе ПЛИС Altera Cyclone EP1C3T144.** Данный прототип предназначен для обучения работе с ПЛИС архитектурой и спроектирован с таким расчетом, чтобы получить плату при минимальных материальных затратах. Применена ПЛИС EP1C3T144, состоящая из 2910 логических ячеек (macrocells), 59904 бита встроенной RAM, присутствует PLL – самый дешевый из семейства Altera Cyclone. Электрическая схема и односторонняя печатная плата представлена на рис. 3 и 4.

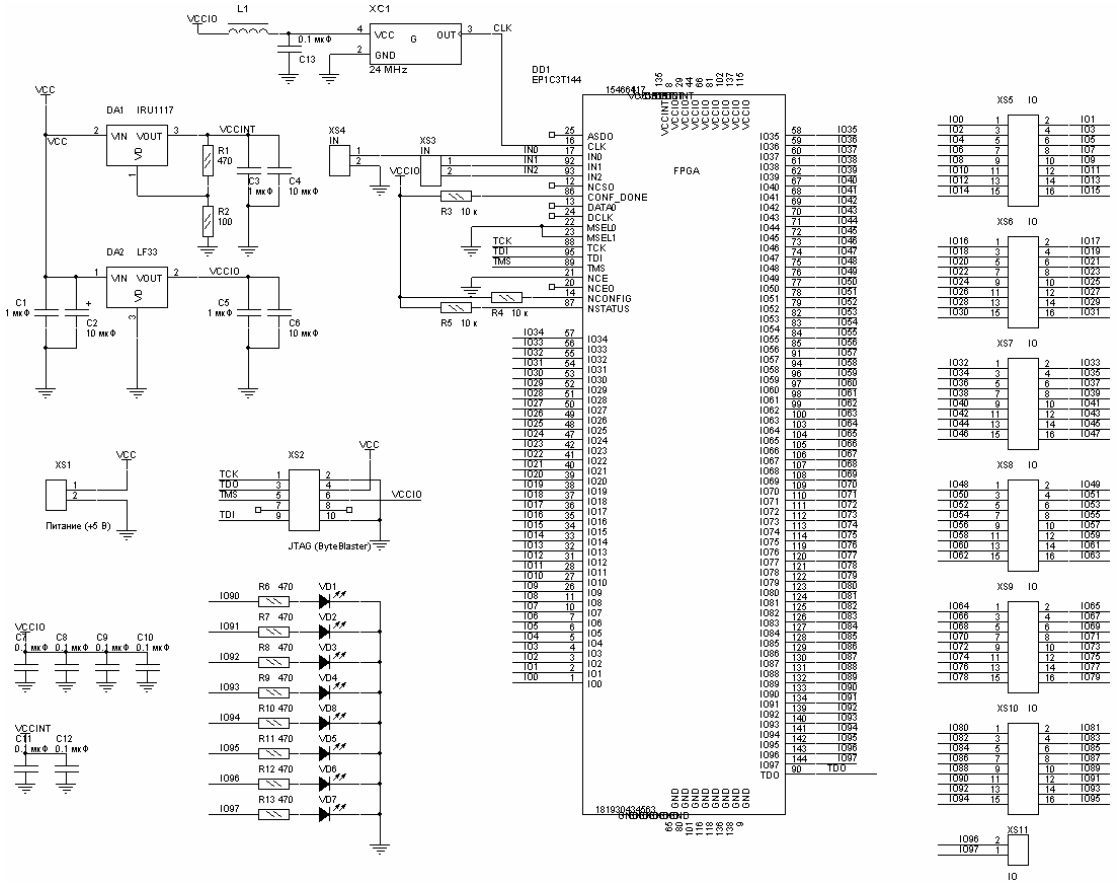



Рис. 3. Электрическая схема прототипа, построенного на основе ПЛИС

Стоит заметить, что печатная плата была разведена с помощью отечественной разработки – авторассировщика TopoR, позволяющего работать в трассировочном режиме freestyle. На эту ПЛИС можно загрузить реконфигурируемый 32-х разрядный RISC процессор NIOS-II. Целесообразно использовать NIOS в минимальном режиме – ядро, встроенная в FPGA RAM-память (данных и программ) и порт ввода-вывода для управления внешними выводами (светодиодами). Процессор в такой конфигурации занимает 20 % кристалла, т.е. еще остается достаточно места для дополнительной периферии и логики. Данный прототип позволит в короткие сроки освоить ПЛИС архитектуру, осуществлять разработку встроенных систем начального уровня сложности, а возможность модульного расширения позволит создавать более сложные системы, в том числе и многопроцессорные.

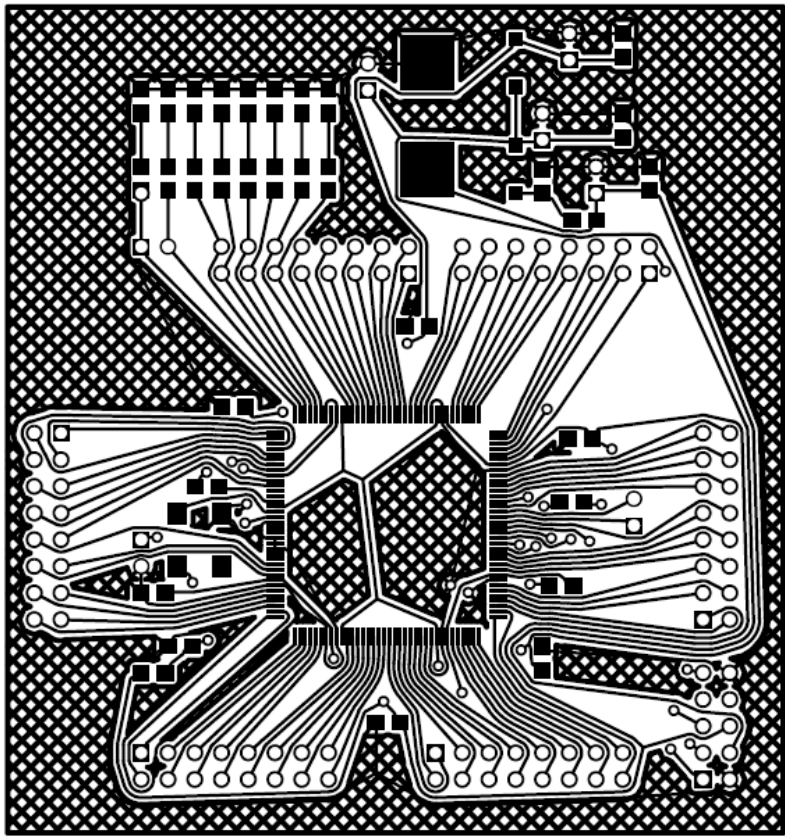


Рис. 4. Топология печатной платы прототипа, построенной на основе ПЛИС

## Заключение

Представленные платы являются производственными прототипами и предназначены для отладки программно-аппаратных решений в составе разрабатываемого автором программно-аппаратного комплекса автоматизированного проектирования встроенных устройств. Возможно применение этих плат в качестве учебного пособия. После минимальной доработкой их можно применять во встраиваемых коммерческих системах.

## Литература


1. www.ti.com
2. http://ru.wikipedia.org/wiki/Verilog
3. http://ru.wikipedia.org/wiki/VHDL
4. Борисевич А.В. Некоторые вопросы эффективного построения проверяющего теста для ПЛИС. // Тезисы докладов Всеукраинской научно-технической конференции «Автоматизация: проблемы, идеи, решения». Севастополь, 16–18 мая 2006 г. Севастополь: изд. СевНТУ, 2006.
5. http://www.terralab.ru/video/234545/
6. http://electronix.ru/forum/index.php?showforum=139

# ПРИМЕНЕНИЕ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ ПРИ ОПТИМИЗАЦИИ ТЕХНИЧЕСКИХ И СХЕМНЫХ РЕШЕНИЙ ХОЛОДИЛЬНЫХ СИСТЕМ

**А.В. Пазухин**
**Научный руководитель – д.т.н., профессор А.Г. Коробейников**

В статье рассматриваются актуальные вопросы создания систем автоматизации проектирования, необходимых для ускорения процессов проектирования холодильных систем. Показана общая принципиальная схема процессов синтеза и оптимизации проектирования систем холодоснабжения, а также приведена предлагаемая к рассмотрению подсистема автоматизации проектирования объектов холодоснабжения.

## Введение

Холодильная техника находит широкое применение на предприятиях пищевой, нефтехимической, нефтеперерабатывающей и других отраслей промышленности, причем затраты на производство умеренного холода составляют значительную часть общей суммы затрат на все технологическое производство (до 25 %). Высокий уровень технического прогресса привел к созданию высокоинтенсивных технологических процессов, агрегатов большой единичной мощности, что предопределило резкое увеличение холодопотребления предприятий.

Производство умеренного холода на целом ряде производств сочетается с процессами низкопотенциального теплоснабжения в соответствующем диапазоне температур, что приводит к дополнительному усложнению систем хладоснабжения. Сложность внутренних взаимосвязей параметров, процессов и характеристик отдельных элементов системы хладоснабжения предопределяет необходимость совершенствования научных, инженерно-технических и технико-экономических решений, обеспечивающих возможность значительного снижения капитальных и энергетических затрат на производство холода и сокращение сроков проектирования. Существенное повышение качества проектирования систем хладоснабжения возможно за счет интенсификации и координирования научно-исследовательских работ, направленных на создание перспективных и надежных систем хладоснабжения и рациональных способов их регулирования, ускорения разработки их математического описания с целью последующего использования в процессах проектирования. При этом возникает необходимость решения научно-технических проблем холодильной техники, связанных с разработкой современных методов автоматизированного проектирования с помощью ЭВМ, обеспечивающих возможность проведения оптимизационных проектных исследований систем хладоснабжения и координирование результатов этих исследований с результатами исследований других подсистем технологического производства с целью достижения оптимальности общего решения [1].

В данной статье в наиболее общем виде изложены процессы синтеза и оптимизации при проектировании холодильных систем, а также предлагается к рассмотрению структура подсистемы, предназначенной для реализации автоматизированного проектирования объектов холодоснабжения.

## Структурная и параметрическая оптимизации проектирования холодильных систем

В настоящее время технически сложные системы холодоснабжения предприятий разрабатываются подразделениями проектных и проектно-конструкторских организаций, при этом методические положения по координации многоуровневых системных процессов автоматизированного проектирования отдельных подсистем практически

отсутствуют. Итерационный подход к согласованию отдельных технических решений в подразделениях приводит к увеличению времени проектирования систем, а также не позволяет получать оптимальные решения без отлаженных координирующих воздействий. В наибольшей степени это касается проектирования нетиповых схем с применением единично разрабатываемого теплообменного и компрессорного оборудования, а также проектирования систем с переменными условиями эксплуатации.

Развитие вычислительной техники создает условия для перехода к новому этапу автоматизации процесса проектирования, а именно к созданию систем автоматизированного проектирования холодильных установок путем сопряжения локальных вычислительных комплексов, обеспечивающих проектирование отдельных узлов и элементов. При этом необходима разработка таких методов структурно-параметрической оптимизации холодильной установки и отдельных ее подсистем, которые обеспечили бы возможность построения единого алгоритма всего процесса проектирования при реализации произвольной задачи как при построении новых, так и при модернизации существующих систем.

Таким образом, появляется необходимость создания скоординированной системы, обеспечивающей возможность не только проектирования, но и оперативной оценки воздействия от реализации любой идеи на эффективность холодильной установки с помощью численного приближенного исследования. Очевидно, что создание такого механизма исследований возможно только при наличии методов структурной и параметрической оптимизации установок с произвольным схемным решением и учетом особенностей эксплуатации и надежности, методов математического моделирования отдельных элементов и элементарных процессов, методов описания свойств рабочих тел, методов автоматизации построения математических моделей сложных систем.

При соответствующем техническом и организационном обеспечении этой системы сроки внедрения разработок значительно сокращаются, резко повышается качество проектных работ. Разработанное методическое и программное обеспечение должно быть использовано не только для проведения проектных исследований, обработки экспериментальных данных, численных экспериментов, но и для создания подсистем для оценки технического состояния и автоматического управления холодильных установок.

Для систем хладоснабжения при условии неизменности результатов задача оптимизации должна сводиться к достижению условия [2]:

$$\min \int_{t} З_t a_t dt = \min \sum_{t=0}^{T} З_t a_t \quad , \qquad (1)$$

где $З_t$ – затраты на реализацию проекта в году t; $T$ – горизонт расчета; $a_t$ – коэффициент дисконтирования.

Случайное воздействие предлагается оценивать с помощью его математического ожидания, что приводит к формальному отсутствию неопределенности и возможности использовать дискретно-непрерывные или детерминированные модели, при этом результаты интерпретируются с учетом вероятностной природы указанных параметров. При наличии информации о статистических законах распределения отказов [3], интенсивности изменения эксплуатационных параметров отдельных элементов и схеме связей между ними появляется возможность определения эффективности работы системы с использованием основных способов повышения надежности (изменения числа температур кипения, резервирования, изменения конструкции элементов и т.п.).

Исходя из этого, (1) можно представить в следующем виде:

$$\min \sum_{i} З_i x_i .$$

где

$$З_t = \sum_i (K_i + K_{рез i}) + \int_t Э_t dt + \int_t C_{ущ_t} dt;$$

$K_i$, $K_{рез i}$ – капитальные затраты на рабочее и резервное оборудование; $C_{ущ}$ – стоимость ущерба от простоя оборудования; $Э$ – эксплуатационные затраты в единицу времени.

В зависимости от назначения, системы хладоснабжения могут рассматриваться как автономные энерготехнологические объекты или объекты, входящие в состав производственно-технологического предприятия.

Как показывает анализ функциональных особенностей холодильных установок в процессе проектных исследований, в большинстве случаев указанное выше выделение систем хладоснабжения в процессе декомпозиции предприятий в самостоятельные объекты исследования оказывается возможным и рациональным при корректных координирующих взаимодействиях отдельных подразделений и проектных организаций. При системном подходе формализация систем хладоснабжения как объекта является базой для построения иерархии и создания методологий проектирования, эксплуатации исследования, она рассматривается как отдельная многоэлементная система с иерархической структурой и как квазистационарная физико-техническая система с большим числом внешних воздействий и внутренних взаимосвязей и ограничений.

Для реализации всестороннего исследования систем хладоснабжения предлагается:

- введение универсальности иерархического принципа декомпозиции с использованием обобщенных функциональных особенностей выделенных подсистем;
- осуществление декомпозиции системы на подсистемы, которые имеют максимальную автономность (минимальное количество связей) как с позиции проведения предварительных экспериментальных и теоретических исследований, так и проведения синтеза и оптимизации непосредственно в процессе проектирования.

На базе указанной формализации предлагается строить:

- иерархию проектно-конструкторских и экспериментально-исследовательских работ;
- методологию структурной и параметрической оптимизации систем хладоснабжения;
- иерархическую структуру математической модели.

Именно сочетание принципов декомпозиции и композиции позволяет решать задачи координирования этапов структурной и параметрической оптимизации основного оборудования, а значит, и предопределяет саму возможность деления процесса оптимизации на отдельные этапы и объединения их в единый алгоритм.

Базируясь на указанных подходах, процесс создания методики структурной и параметрической оптимизации систем хладоснабжения можно представить в виде укрупненной структуры, показанной на рис. 1.

Структурно-параметрическая оптимизация систем хладоснабжения, таким образом, сводится к последовательному синтезу структур и оптимизации параметров отдельных комплексов (подсистем) с привлечением методов сопряжения с другими комплексами (подсистемами) на базе имитационного (регрессионного) моделирования. Имитационные (регрессионные) модели подсистем и элементов представляет собой информационные отражения в детерминированном или индетерминированном виде связи между обобщенными входными параметрами и оптимальным откликом, которые характерны только для конкретной подсистемы с конкретными стоимостными характеристиками используемых материалов, оборудования и энергии.

Откликом в данном случае являются значения обобщенных физических характеристик (перепады температур, давлений и др.). Зависимости для расчета оптимальных технико-экономических показателей также включаются во множество имитационных соотношений, а результаты – во множество параметров отклика.

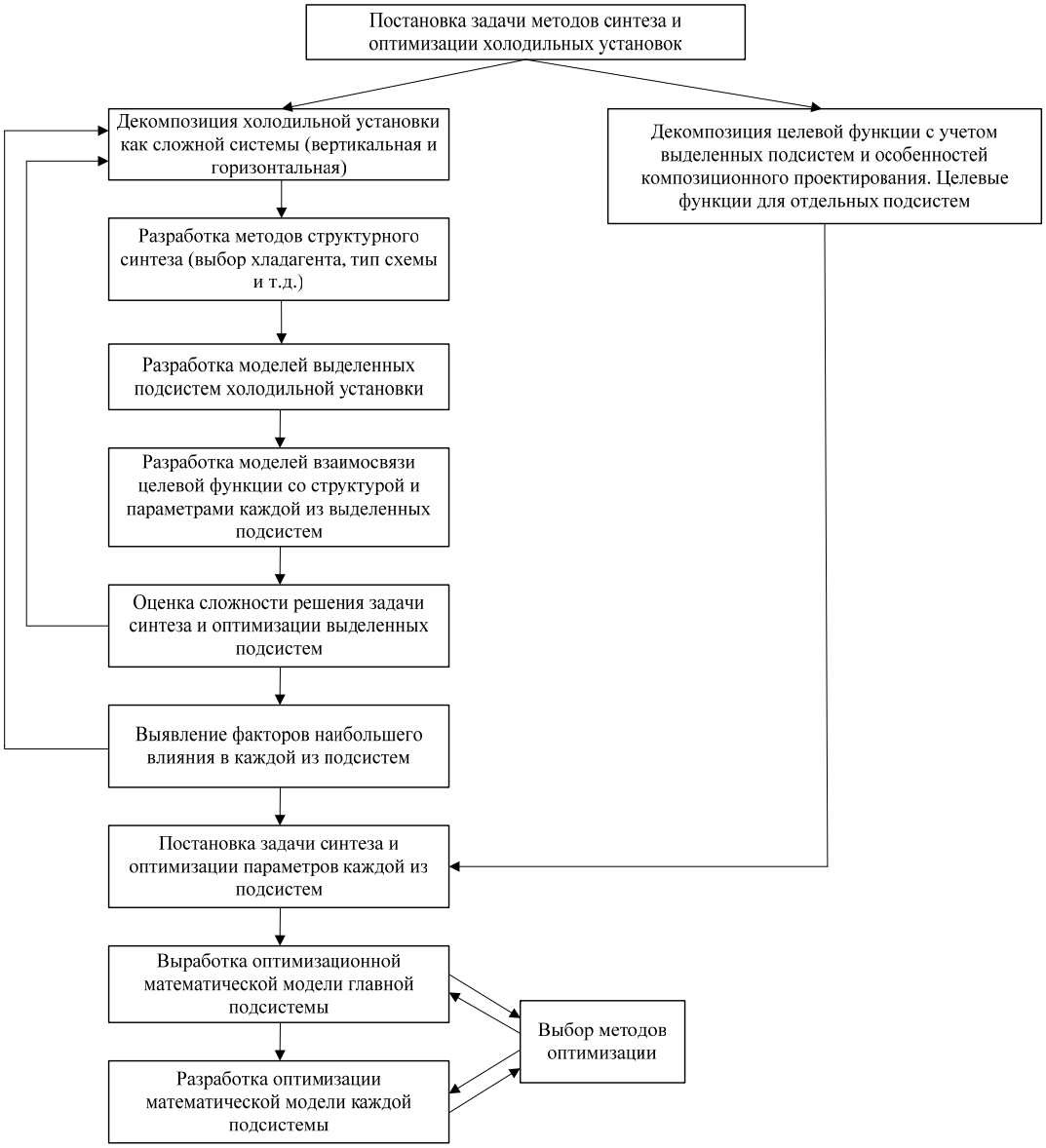



Рис. 1. Принципиальная структурная схема процесса разработки методов синтеза и оптимизации холодильной установки

Сформулированная задача структурно-параметрической оптимизации является весьма сложной нелинейной задачей, а так как возникает необходимость оптимизировать состав оборудования, то указанная задача приобретает комбинаторно-дискретный характер. Достаточно корректное решение может быть получено только при глубоком анализе специфики задачи и использовании методов многоуровневой оптимизации [4].

## Описание реализации

Исходя из описанного выше, можно заключить, что обобщенный алгоритм структурно-параметрического синтеза холодильных установок следует формировать из нескольких скоординированных этапов:

- выбор хладагентов и типа схемы;
- направленный поиск приближенной структуры и оптимального распределения потребителей холода и тепла по температурным уровням (или диапазонам) при использовании имитационного (регрессионного) моделирования теплообменного и компрессорного оборудования;

- эмпирико-эвристический последовательный синтез уточненных структур основного энергетического комплекса с вариантами фиксированных множеств температур кипения хладагента на базе соотношений о степени воздействия изменения структуры на общую эффективность системы;
- структурно-параметрическая оптимизация с формированием математических моделей.

На рис. 2 показана структурная схема подсистемы энергетического комплекса, разработанная на основе структурно-параметрического синтеза и предназначенная для реализации автоматизированного проектирования сложных холодильных систем с большим числом элементов.

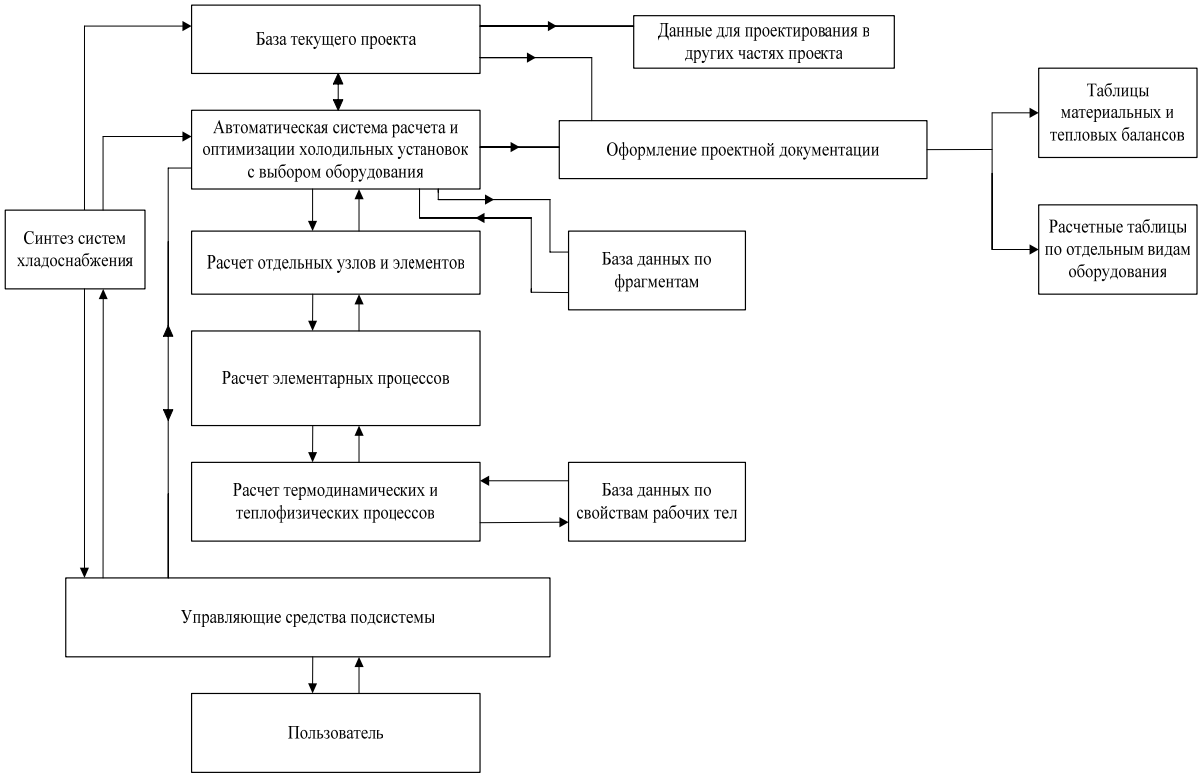



Рис. 2. Структурная схема реализации автоматизированного проектирования холодильной системы

Внедрение рассмотренной системы при проектировании систем холодоснабжения позволит:

- подобрать наиболее оптимальное в конкретных условиях оборудование, а также оптимальное соотношение взаимосвязанного между собой оборудование различных типов (компрессорное – теплообменное, теплообменное – насосное и т.д.);
- уменьшить общее время прохождения проекта внутри проектной организации от составления технического задания до подготовки готового комплекта проектной и/или рабочей документации;
- скоординировать действия отделов при разработке технических решений;
- установить степени ответственности за принятие технических решений и время разработки отдельной части общего энергетического комплекса внутри подразделения;
- минимизировать вероятность отдельных ошибочных действий в процессе проектирования и предотвратить их негативное влияние.

## Заключение

Для реализации поставленных в статье вопросов необходима разработка системы автоматизированного проектирования на базе предложенной выше схемы, отвечающая современным потребностям конструкторских подразделений проектных организаций и основанная на современных разработках как вычислительной техники, так и холодильного машиностроения.

## Литература


1. Курылев Е.С., Оносовский В.В., Румянцев Ю.Д. Холодильные установки. СПб. Политехника, 2002.
2. Методические рекомендации по оценке эффективности инвестиционных проектов и их отбору для финансирования. Утверждено Госстроем России, № 7-12/47, 31 марта 1994 г. / Стройинформ-СПб. Спец. выпуск. 1995.
3. Петров Е.Т., Лукьянова Т.А. Обработка статистической информации по надежности отдельных элементов компрессорных станций. Деп. сб. "Новые исследования холодильных машин и установок". ЦИНТИхимнефтемаш, № 1, 1985.
4. Петров Е.Т., Михновская Е.Л. Оптимизация холодильных установок в процессе автоматизированного проектирования. / Автоматизированное проектирование трубопроводных систем нефтеперерабатывающих и нефтехимических производств. М.: ЦНИИЭнефтехим, 1982.

# АВТОМАТИЗИРОВАННАЯ СИСТЕМА ВЫЯВЛЕНИЯ ЭЛЕКТРОМАГНИТНЫХ КРАТКОСРОЧНЫХ ПРЕДВЕСТНИКОВ СИЛЬНЫХ ЗЕМЛЕТРЯСЕНИЙ НА ОСНОВЕ ГЕОФИЗИЧЕСКОЙ ИНФОРМАЦИИ

**Д.Ю. Сарычев**
**Научный руководитель – д.т.н., профессор А.Г. Коробейников**

Рассматриваются принципы построения и работы автоматизированной системы, выявляющей электромагнитные краткосрочные предвестники сильных землетрясений на основе анализа геофизической информации – определения векторов градиента и фазовых скоростей ультранизкочастотных геомагнитных возмущений вдоль земной поверхности.

## Введение

Исследования градиентов и фазовых скоростей ультранизкочастотных (УНЧ) геомагнитных вариаций перед землетрясениями 2000 и 2002 годов возле полуострова Изу и на полуострове Босо (юго-западнее и юго-восточнее Токио) показало, что примерно за 3–6 месяцев до сильных землетрясений (магнитуда больше 5) начиналось аномальное увеличение величин градиентов в вертикальной и полной горизонтальной компонентах магнитного поля и уменьшение величин фазовой скорости в этих же компонентах [1–4]. Было высказано предположение о том, что аномальное изменение градиентов и фазовых скоростей связано с двумя процессами в области очага будущего землетрясения – образуется аномалия повышенной проводимости, и возникают широкополосные литосферные УНЧ электромагнитные излучения. Аномалия повышенной проводимости может возникнуть как вследствие тектонических движений, так и вследствие подъема магмы к поверхности земной коры. Один из возможных механизмов возникновения электромагнитных излучений связан с активизацией процесса образования микротрещин в области очага будущего землетрясения [5]. Высокочастотные электромагнитные излучения сильно затухают в земной коре, и на поверхности мы наблюдаем, в основном, ультранизкочастотные излучения ($F < 1$ Гц).

Таким образом, измерение и анализ электромагнитных волн в УНЧ-диапазоне позволяет сделать краткосрочный прогноз местоположения предстоящего очага и времени землетрясения.

## Методы измерения и анализа электромагнитных волн

Градиенты и фазовые скорости геомагнитных вариаций можно находить двумя способами – чисто экспериментальным путем и в рамках модели плоской электромагнитной волны. В первом случае необходимо определять фазовые задержки и разности величин амплитуд вариаций между двумя любыми парами станций магнитного градиентометра, состоящего из трех разнесенных станций. Поскольку координаты магнитных станций и расстояние между ними известно, можно определить градиенты и фазовые скорости для двух пар станций, выбранных из трех станций магнитного градиентометра, и затем построить вектор фазовой скорости и градиента пульсаций в соответствии с формулами, приведенными в [3] и [6]. Поскольку фазовые скорости геомагнитных волн вдоль земной поверхности для УНЧ геомагнитных вариаций составляют десятки км/с, а фазовые задержки, соответственно, доли секунды, то необходимо использовать данные с дискретностью 50 Гц. В рамках модели плоской электромагнитной волны величина фазовой скорости между двумя точками на земной поверхности определяется через амплитуды соответствующих компонент вариаций магнитного поля с учетом фазовой задержки [6, 7] следующим образом:

$$V_{ij} = \frac{2\pi d_{ij}}{T \ln \frac{B_i(t)}{B_j(t+\tau)}}.$$

В этом выражении для геомагнитных вариаций с периодом $T$ величины $B_i$ и $B_j$ определяются в момент времени $t$ на первой станции и в момент времени $t+\tau$ на второй станции ($\tau$ – фазовая задержка при прохождении геомагнитной волны расстояния $d_{ij}$ между двумя станциями).

Используя данные, полученные от трех станции, расположенных на земной поверхности в виде треугольника, можно в соответствии с вышеприведенной формулой определить фазовые скорости $V_{12}$ (между станциями 1 и 2) и $V_{13}$ (между станциями 1 и 3) и затем найти направление и величину вектора фазовой скорости геомагнитных волн вдоль земной поверхности. При применении этого метода нет необходимости в высокой дискретности регистрируемых данных, поскольку в качестве величин $B_i$ и $B_j$ могут быть использованы среднеквадратичные значения амплитуд УНЧ геомагнитных пульсаций.

## Принципы построения и работы автоматизированной системы выявления электромагнитных краткосрочных предвестников сильных землетрясений

На рис. 1 показана блок-схема предлагаемой автоматизированной системы сбора и обработки геофизической информации. Группы станций 1 и 2 состоят каждая из трех трехкомпонентных магнитовариационных станций, расположенных на земной поверхности в вершинах треугольника на расстоянии 3–5 км друг от друга. Расстояние между двумя группами станций составляет 80–100 км, а расстояние от них до ЦСИО – 100–1000 км. При такой конфигурации установленных датчиков будет контролироваться район 200×200 км. Данные, регистрируемые каждой магнитной станцией с дискретностью 50 Гц, поступают в ЦСИО каждые три часа.

В результате обработки на экран компьютера выводятся следующие данные:

- среднеквадратические значения геомагнитных вариаций в полосе частот 0.001-1 Гц, разбитой на 10 поддиапазонов;
- величины векторов градиентов и фазовых скоростей геомагнитных вариаций в той же полосе частот;
- направления векторов градиентов и фазовых скоростей геомагнитных вариаций в той же полосе частот.

В случае появления устойчивых новых направлений векторов градиентов и фазовых скоростей, а также аномального изменения величин градиентов и фазовых скоростей производится расчет местоположения будущего эпицентра землетрясения и выдается сигнал тревоги.

## Использование системы

Разработанная автоматизированная система сбора и обработки геофизической информации для выявления электромагнитных краткосрочных предвестников сильных землетрясений с 2000 г. активно эксплуатируется в Японии (южнее Токио). На рис. 2 показано расположение станций (треугольники) и эпицентров землетрясений (звездочки) в 2000 и 2002 годах, цифры означают магнитуды землетрясений. В этом районе ведется регистрация вариаций магнитного поля и теллурических токов шестью высокочувствительными цифровыми трех компонентными станциями MVC-3DS, разработанными в Санкт-Петербургском филиале института земного магнетизма, ионосферы и распространения радиоволн РАН. Три станции расположены на полуострове Идзу, и

три – на полуострове Босо. Каждая группа станций представляет собой магнитный градиентомер, расстояние между станциями в группе составляет 4–7 км.


Рис. 1. Блок-схема автоматизированной системы сбора и обработки геофизической информации в реальном масштабе времени

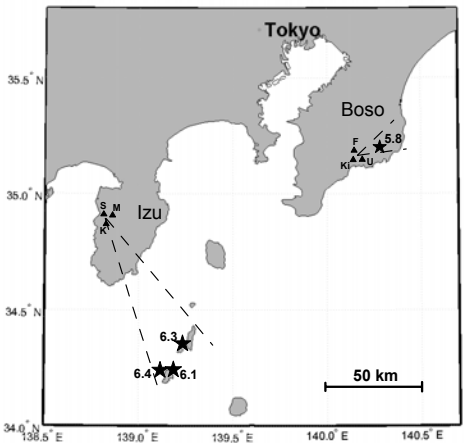



Рис. 2. Расположение магнитных станций (черные треугольники) и эпицентров сильных землетрясений (звездочки) в Японии в 2000 (Изу) и 2002 (Босо) годах

Данные, регистрируемые каждой магнитовариационной станцией, записываются на жесткий диск ПК с дискретностью 50 Гц; для синхронизации одновременной работы магнитных датчиков используется система GPS. Регистрируемые данные с дискретностью 1 Гц ежесуточно передаются в центр сбора и обработки информации, расположенный в университете города Тиба.

## Литература


1. Kawate R. Ultra-low-frequency magnetic fields during the Guam earthquake of 8 August 1993 and their interpretation. // Phys. Earth Planet. Interiors. 1998. V. 105.
2. Goto T.-N. Calibration and running test of torsion magnetometer made in Russia. // Rep. of Japan Marin Sci. and Tech. Center (JAMSTEC), 2002. V. 45.
3. Kopytenko Yu.A. Investigation of the ULF electromagnetic phenomena related to earthquakes: contemporary achievements and the perspectives. // Annali di Geofisika. 2001. V. 44. № 2.
4. Kopytenko Yu.A. Monitoring of the ULF electromagnetic disturbances at the station network before EQ in seismic zones of Izu and Chiba peninsulas. / In: Seismo Electromagnetics: Litosphere-Atmosphere-Ionosphere Coupling. Eds. M. Hayakawa and O.A. Molchanov. Tokyo: TERRAPUB, 2002.
5. Molchanov O.A. Generation of ULF electromagnetic emissions by microfracturing. // Geoph. Res. Lett. 1995. V.22.
6. Ismaguilov V.S. Variations of phase velocity and gradient values of ULF geomagnetic disturbances connected with the Izu strong earthquakes. // Natural Hazards and Earth Sys. Sci. 2002. V.20.
7. Ismaguilov V.S. ULF Magnetic Emissions Connected with Under Sea Bottom Earthquakes. // Natural Hazards and Earth Sys. Sci. 2001. V.1.

# ПРОЕКТИРОВАНИЕ КОРПОРАТИВНОЙ АВТОМАТИЗИРОВАННОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ «БАНКОМАТ +»

**В.В. Соловьёв**
**Научный руководитель – д.т.н., профессор В.Л. Ткалич**

В работе рассмотрена возможность создания автоматизированной информационной системы, охватывающей все аспекты работы банкоматов. Рассмотрены возможности программ-аналогов. Построена структурная схема проектируемой системы. Приведено описание возможностей автоматизированной системы, особенностей ее создания и и внедрения..

## Введение

Одним из наиболее эффективных и экономичных устройств для предоставления розничных банковских услуг является банкомат (АТМ). Банкомат позволяет держателю карточки получать информацию о текущем состоянии счета и проводить операции по перечислению средств с одного счета на другой. Банкоматная сеть представляет собой совокупность АТМ, устанавливаемых в филиалах банка, торгово-сервисных предприятиях или на территории корпоративных клиентов банка, и каналов передачи данных, связывающих терминальные устройства с процессинговым центром банка. При этом качество обслуживания клиентов банка в банкоматной сети определяется, с одной стороны, составом и содержанием предлагаемых банком услуг, с другой стороны, их доступностью. Доступность предлагаемых услуг напрямую зависит от численности и разветвленности банкоматного парка, режима работы банкоматов, а также от качества работы сети банкоматов в целом. Степень удовлетворенности клиента может быть повышена путем установки дополнительного «избыточного» числа банкоматов, однако этот путь требует значительных затрат. Количественно степень удовлетворенности клиента может быть определена как вероятность получения услуги при обращении к АТМ. На этот показатель влияют частота и длительность периодов неработоспособности банкоматов. Под неработоспособностью банкомата подразумевается отсутствие по тем или иным причинам возможности обеспечивать банкоматом выполнение основной функции – выдачи наличных денежных средств [1]. Причинами, приводящими к неработоспособности банкомата, могут являться:

- отказы технических средств и программного обеспечения;
- неисправность канала связи;
- отсутствие ресурсов расходных материалов (журнальной и чековой ленты) и наличных денежных средств.

Для сокращения времени простоя банкоматов создаются различные автоматизированные информационные системы. Информация, которую приходится учитывать при работе банкомата, весьма разнообразна. Соответственно, различны и задачи, которые решаются автоматизированными системами. В настоящее время существует большое количество автоматизированных информационных систем для учета проводимых мероприятий по обслуживанию банкоматов. Но не все банки приобретают уже готовые программные продукты, предпочитая разрабатывать подобные корпоративные автоматизированные информационные системы своими силами. Но и готовое, и «собственное» ПО ориентировано на решение лишь некоторых задач. Возможно ли создать такую универсальную автоматизированную информационную систему, которая охватывала бы не только все вопросы, возникающие при работе банкоматов, но и всю деятельность, связанную с ними?

## Обзор аналогов

**ATM-Analyst.** Автоматизированная система «ATM-Analyst» предназначена для решения задач контроля и анализа показателей работоспособности сети банкоматов. Целью внедрения автоматизированной системы «ATM-Analyst» является сокращение времени простоя банкоматов. В состав автоматизированной системы «ATM-Analyst» входят следующие компоненты:

- входные файлы (от системы мониторинга сети банкоматов и системы сервисного обслуживания сети банкоматов);
- хранилище данных;
- сервер базы данных;
- сервер безопасности;
- база данных безопасности;
- OLAP-сервер;
- средства загрузки хранилища данных;
- модуль «Администратор»;
- модуль «Аналитик».

В соответствии с устанавливаемым регламентом на вход автоматизированной системы «ATM-Analyst» поступают входные файлы от систем мониторинга и сервисного обслуживания сети банкоматов за период времени (например, за сутки). Эти входные данные с помощью средств загрузки записываются в хранилище данных. Хранилище данных в результате содержит всю необходимую информацию для решения задач контроля и анализа работы сети банкоматов.

Для обеспечения доступа к аналитической информации используется OLAP-сервер (On-line Analytical Processing сервер). Работа пользователя категории «системный аналитик» с аналитической информацией осуществляется с помощью модуля «Аналитик», обеспечивающего представление аналитической информации в форме таблиц, графиков, диаграмм. Обеспечение требований по безопасности при функционировании автоматизированной системы «ATM-Analyst» осуществляется с использованием сервера безопасности, базы данных безопасности и модуля «Администратор». Сервер безопасности обеспечивает решение следующих задач: введение пользователей в систему, аутентификация пользователей, аудит выполняемых пользователей операций. Данные по пользователям, журнал аудита, параметры настройки автоматизированной системы «ATM-Analyst» хранятся в базе данных безопасности. Модуль «Администратор» обеспечивает диалоговое взаимодействие с пользователем категории «администратор» при изменении параметров настройки автоматизированной системы «ATM-Analyst» и ведении нормативно-справочной информации в хранилище данных [2].

**ATM-Monitor.** Автоматизированная система «ATM-Monitor» предназначена для автоматизации деятельности персонала банка и сервисных компаний, участвующих в процессе эксплуатации сети банкоматов.

Целью создания автоматизированной системы «ATM-Monitor 1.0» являются улучшение показателей качества работы сети банкоматов за счет:

- оперативного выявления проблемных ситуаций в сети банкоматов;
- оперативного формирования заявок на сервисное обслуживание;
- контроля выполнения заявок на сервисное обслуживание.

Система «ATM-Monitor» обеспечивает выполнение следующих основных функций:

- графическую визуализацию технического состояния банкоматов в заданном масштабе времени;
- индикацию фактов неработоспособности банкоматов в разрезе причин неработоспособности;
- выдачу детальной информации по факту неработоспособности;

- контроль показателей качества работы сети банкоматов;
- формирование заявки на сервисное обслуживание в сервисную компанию;
- контроль выполнения заявки на сервисное обслуживание.

Система «ATM-Monitor» может осуществлять взаимодействие с системой сервисного обслуживания «БИТ: Сервисное предприятие», обеспечивая передачу заявок на сервисное обслуживание и прием данных о выполнении заявок [2].

**БИТ: Сервисное предприятие.** Автоматизированная система «БИТ: Сервисное предприятие» предназначена для автоматизации бизнес-процессов малых и средних предприятий сферы услуг, обеспечивающих:

- сервисное обслуживание контрольно-кассовых машин, транзакционного оборудования (банкоматы, торговые терминалы), средств вычислительной техники и другого оборудования;
- услуги телефонной связи;
- коммунальные услуги и многое другое.

Автоматизированная система «БИТ: Сервисное предприятие» состоит из трех компонентов:

1. комплекс программ «Договор»;
2. комплекс программ «Сервис»;
3. компонента «1С: Бухгалтерия 7.7: Конфигурация Сервисное обслуживание».

Компонента «Договор» обеспечивает выполнение следующих функций:

- ведение договоров предприятий как в целом по организации, так и по внутренним подразделениям;
- автоматический расчет стоимости работ/услуг, оказанных за выбранный период;
- групповое или выборочное формирование счетов по контрагентам, договорам, исполнителям и типам работ;
- учет взаиморасчетов с заказчиками;
- формирование аналитических отчетов о заключенных договорах, состоянии оборудования, дебиторах, полученных и планируемых доходах и т.д.;
- информационное взаимодействие с программой «1С: Бухгалтерия 7.7: Конфигурация Сервисное обслуживание».

Компонента «Сервис» обеспечивает выполнение следующих функций:

- регистрация заявок на сервисное обслуживание;
- планирование и учет выполнения работ по заявкам на сервисное обслуживание;
- учет выполнения заявок на сервисное обслуживание;
- учет израсходованных материалов и комплектующих;
- учет отказов оборудования с детализацией по категориям дефектов и отказавшим узлам;
- выставление счетов за израсходованные детали и дополнительные работы и, не предусмотренные договором;
- получение аналитических отчетов о состоянии оборудования, выполнении заявок, оказанных услугах, использовании транспортных средств, отказах оборудования и т.д.

Компонента «1С: Бухгалтерия 7.7. Конфигурация Сервисное предприятие» обеспечивает возможность:

- группового или выборочного формирования счетов, актов оказания услуг и счетов-фактур в любой валюте;
- автоматической выгрузке за указанный период оплат за работы/услуги из программы «1С Бухгалтерия 7.7»;
- приема информации о контрагентах, договорах, работах и номенклатуре обслуживаемых изделий;
- выгрузки данных об ассортименте и количестве материалов и комплектующих на складах;

- автоматического формирования документов по оплате коммунальных услуг и междугородних/международных переговоров [2].

## Автоматизированная информационная система «Банкомат +»: ее структура и возможности

Как уже говорилось, что не все банки приобретают готовые программные продукты, предпочитая разрабатывать подобные автоматизированные системы своими силами. Непосредственной разработкой ПО занимается головной офис банка, решая определенные задачи для процессинга, и может предоставить филиалам, как показано на рисунке, только мониторинг технического состояния их банкоматов. К сожалению, филиалам для решения своих задач по обслуживанию банкоматов порой не хватает возможностей ПО, разработанного в головном офисе банка. Создание автоматизированной информационной системы «Банкомат +» позволит решить данные задачи и оптимизировать ресурсозатраты по обслуживанию банкоматов. На рис. 1 изображена схема подключения проектируемой системы в работу банкоматной сети. По каналам связи ведется передача данных от банкоматов в процессинговый центр. Из головного офиса в технический отдел передается информация о состоянии банкоматов. Разрабатываемая система должна устанавливаться в филиале и подключаться по другому каналу связи напрямую к банкоматам. На каждом банкомате с помощью всевозможных технических средств разрешается доступ к его информационным ресурсам не только с головного офиса, но и из филиала банка. В частности, разрешается доступ к файлам электронного журнала, которые и будут служить входными параметрами для автоматизированной информационной системы.

Целью создания данной автоматизированной системы являются:

- сокращение времени простоя в работе банкоматов;
- мониторинг состояния банкоматов в реальном времени;
- автоматическое планирование профилактических работ по техническому обслуживанию;
- ведение статистики по выполненным работам;
- прогнозирование появление неисправностей в работе банкоматов;
- возможность экспертной помощи техническим специалистам в вопросах неисправностей банкоматов;
- возможность планирования маршрутов проезда по городу для оптимизации трудозатрат;
- эффективное взаимодействие между отделами по вопросам работы банкоматов.

Анализ используемых технологий разработки программного обеспечения показал, что наиболее совершенная и прогрессивная технология – это «клиент-сервер», с использованием возможности WEB/database. WEB предлагает стандартизацию пользовательского интерфейса, возможность совместной работы приложений от разных платформ, простоту разработки приложений, легкость поддержки, хорошо стандартизированные отношения «клиент-сервер», возможность использования Интернета. Одновременно базы данных предлагают мощный метод упорядочения и сопровождения информации, представляемой на WEB-страницах, возможность использовать для поиска информации SQL-сервер [3]. Использование этих технологий при создании проектируемой системы позволит быстро и качественно реализовать задуманное.

Автоматизированная информационная система «Банкомат +» включает в себя шесть основных частей (рис. 1):

- базу данных;
- экспертный модуль для технического обслуживания;
- модуль прогнозирования;

- мониторинг состояния банкоматов;
- модуль автоматизированного планирования маршрутов проезда;
- автомобильный мониторинг.

База данных будет хранить всю информацию – от серийных номеров каждого устройства до количества купюр, загружаемых в каждую кассету каждого банкомата.

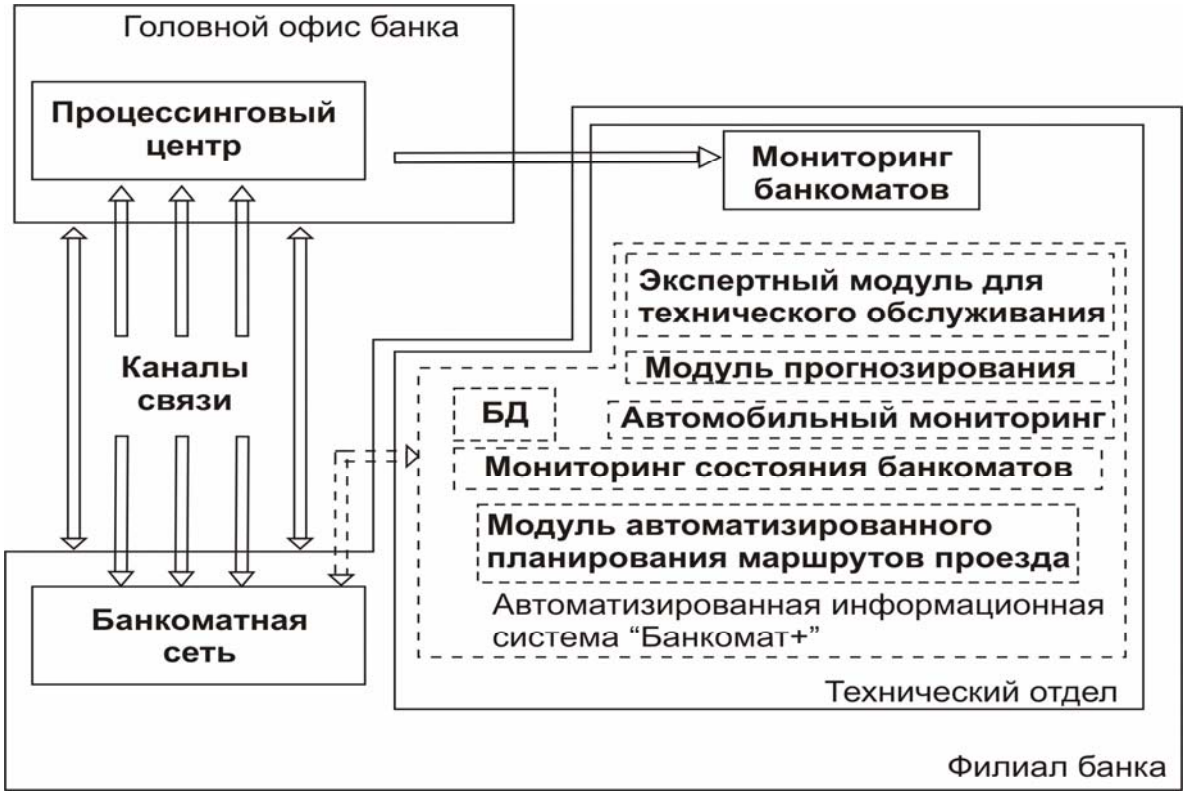



Рис. 1. Схема подключения автоматизированной системы "Банкомат +" в работу и ее основные модули

Экспертный модуль для технического обслуживания позволит техническим специалистам добавлять информацию о ранее неизвестной поломке и ее решении, а также быстро находить причину уже встречавшихся неисправностей в работе банкоматов. Создание такого модуля позволит частично решить вопрос о повышении квалификации технических специалистов, а также даст возможность техническому отделу не потерять основных знаний по решению проблем банкоматов в связи с уходом в отпуск или увольнением ведущего специалиста.

Модуль прогнозирования позволит ориентироваться в сроках окончания денежных средств и расходных материалов, а также предупреждать о возможном появлении неисправности на конкретном банкомате. Данный модуль будет взаимодействовать с модулем автоматизированного планирования маршрутов проезда и модулем автомобильного мониторинга. В свою очередь, модуль планирования маршрутов позволит оптимизировать ресурсозатраты на инкассацию и техническое обслуживание банкоматов. А модуль автомобильного мониторинга в совокупности со специальным оборудованием и рабочим местом оператора позволит следить за местонахождением автомобилей и предупреждать водителей о возникших на дорогах пробках, корректируя маршрут. Взаимодействие этих трех модулей значительно сократит временные затраты на выполнение поставленных задач, связанных с перемещением по городу.

Модуль мониторинга состояния позволит динамически отслеживать работу банкоматов и своевременно реагировать на появление неисправности. Взаимодействие данного модуля с модулем прогнозирования также будет помогать выявлять неста-

бильную работу каждого банкомата, что позволит сократить временные затраты на внеплановые проверки.

## Заключение

Создание автоматизированной системы «Банкомат +» решит много повседневных проблем работников технического отдела банка, а также позволит оптимизировать ресурсозатраты и учитывать всю информацию, окажет незаменимую помощь в оформлении всех отчетов по работе банкоматов.

## Литература


1. Авербух О.В. Качество работы банкоматной сети: как и чем его измерить? //Журнал «ПЛАС. Платежи, системы, карточки». 2004. №1.
2. http://www.bit-it.ru/
3. Гаврилова Т.А., Хорошевский В.Ф. Базы знаний интеллектуальных систем. СПб: ПИТЕР, 2000. 384 с.

# ПРОБЛЕМА АВТОМАТИЗИРОВАННОГО ПЛАНИРОВАНИЯ МАРШРУТА ПРОЕЗДА ПО ГОРОДУ ДЛЯ ОПТИМИЗАЦИИ РЕСУРСОЗАТРАТ

**В.В. Соловьёв**

**Научный руководитель – д.т.н., профессор В.Л. Ткалич**

В работе рассматривается проблема передвижения автотранспорта по большому городу в условиях пробок. Проводится обзор возможностей программных решения автоматизированного планирования маршрутов. Также рассматриваются основные принципы построения данных систем и механизмы планирования маршрутов. Делается вывод о разделении подобных систем на классы по функциональным возможностям.

## Введение

Проблема пробок волнует жителей многих больших городов. Для решения проблемы пробок ставится вопрос о введении зоны платного въезда. Предполагается, что проблемы больших городов, возникшие из-за избытка автотранспорта на улицах, можно решить, если учредить в городе зоны с ограничением на въезд. Возможно, это позволит снизить объемы движения. В частности, такой режим предлагается ввести для некоторых участков центра города и для территорий, расположенных вдоль дорог с интенсивным движением. Ограничить доступ в эти зоны предлагается двумя путями: либо ввести систему пропусков, либо пойти по схеме учета номерных знаков, которая разрешала бы водителям с определенными номерными знаками использовать свои автомобили в определенные дни. Кроме того, предлагается пересадить как можно больше людей на общественный транспорт. Развитие системы муниципальных платных парковок приведет к тому, что в городе будет просто негде бесплатно поставить машину. Все это существенно ограничивает свободу передвижения для владельцев транспортных средств. Для борьбы с пробками нужна комплексная программа, а предлагаемые сегодня законопроекты решают лишь частные вопросы [1]. Пока органы власти пытаются решать проблему пробок на дорогах на уровне законов, многие торговые компании решают этот вопрос приобретением и внедрением у себя автоматизированных программных продуктов для планирования маршрутов проезда по городу с учетом возникающих пробок.

Торговые компании, стремясь не отстать от конкурентов и поддерживать сервис на необходимом уровне, одна за другой объявляют о предоставлении услуги доставки своей продукции. Кому, как не им, известно, что транспортировка в условиях мегаполиса – это риски для эффективности бизнеса и репутации фирмы. Успешная доставка купленного товара зависит в большей степени от погодных условий, технического состояния и загруженности дорог, наличия удобного места парковки. Но, пожалуй, самое главное зависит от правильно составленного маршрута движения транспортного средства с учетом особенностей каждого клиента.

При формировании маршрута необходимо учитывать время прибытия, специфику используемой марки автотранспорта (в соответствии с месторасположением объекта, подъездными путями и типом погрузочно-разгрузочного места), а также состав экипажа маршрута (в соответствии со свойствами перевозимого груза). Если в компании работают опытные планировщики, эта работа с достаточной степенью точности может быть выполнена и вручную. Однако в крупных фирмах, осуществляющих тысячи доставок в день, планируется 200 и более маршрутов в день, а в таких условиях человеческий фактор неизбежно приводит к ошибкам. Это может быть связано как с ошибками в планировании, так и с принятием неэффективных решений из-за отсутствия необходимой информации в момент поступления сигнала о возникшей проблеме (скажем, о пробке на дороге, поломке автомашины или отсутствии клиента по указанному адресу) [2].

Отсутствие необходимой информации, например, о пробке на дороге может обернуться ошибками в планировании маршрутов. Очевидно, что вне зависимости от причин невыполненная доставка, во-первых, весьма негативно скажется на отношении клиента к услугам компании, а во-вторых, приведет к значительным финансовым потерям фирмы, так как потребуется почти полностью повторить цикл обработки заказа.

Эффективным путем избежания описанных проблем является применение систем автоматизированного планирования и мониторинга процесса доставки, которые на этапе формирования маршрута позволяют свести к минимуму негативное влияние человеческого фактора, а с момента выхода машины в рейс – облегчить диспетчеру задачу принятия управленческих решений за счет визуализации места текущего расположения транспортных средств. В настоящий момент на российском рынке программного обеспечения предлагается широкий спектр таких систем.

## Программные решения

**Система мониторинга и навигации транспорта «ANTOR MonitorMasterTM».** ANTOR MonitorMasterTM предназначен для мониторинга и навигации транспорта и грузов, определения отклонений от заданных маршрутов и графиков их передвижения. В состав комплекса входит бортовое устройство и комплекс программных средств для обработки данных и подготовки отчетов. ANTOR MonitorMasterTM обеспечивает сбор и хранение информации о местоположении и состоянии транспорта, грузов и других мобильных объектов с помощью GPS и передачу ее с заданной периодичностью с помощью GPRS-соединения через Интернет.

Программа предоставляет возможность задавать специальные или запрещенные зоны, при этом диспетчеру (менеджеру) автоматически поступает информация в случае, если какой-либо объект пересекает границу таких зон. Специальные фильтры баз данных позволяют диспетчеру (менеджеру) регулировать количество отображаемых объектов в соответствии с заданными параметрами (например, отображать только стратегически важные объекты), что особенно важно, когда диспетчеру приходится отслеживать более десяти объектов одновременно. Параметры настроек могут гибко изменяться в соответствии с бизнес-процессами клиента. Все данные за прошедшие дни попадают в архив и могут быть использованы для проведения дальнейшего анализа.

Вся информация, поступающая в программу со всех устройств, а также планы рейсов накапливаются в базе данных и затем архивируются. Архивация производится сервером автоматически в нерабочее время (например, в полночь). В архиве сохраняются все параметры фактического передвижения автомобиля и все параметры плана. В дальнейшем оператор может выбрать любые рейсы по дате, автомобилям, водителям и загрузить их для просмотра. Впоследствии можно провести анализ рейсов за период, сразу выделить рейсы с отклонениями от заданных параметров и затем подробно изучить их [3].

Данные, сохраненные в архиве, могут быть использованы для статистического анализа рейсов и исполнения плана. Существует возможность проведения анализа работы одной автомашины или группы машин, а также всего объема рейсов сразу. Анализируются такие параметры, как пробег, время в пути, число остановок, число опозданий, количество обслуженных клиентов, средняя скорость, время нахождения у клиентов и т.д. Можно анализировать расчетные и фактические значения, а также разницу между планом и фактом. Программа поддерживает несколько рабочих мест операторов, предназначена для работы с большим количеством автомобилей и не требует постоянного участия человека.

ANTOR MonitorMasterTM работает с двумя типами устройств: стандартный (стационарный) комплект (монтируется в автомобиль и подключается к системе электро-

питания), а также переносное решение (автономное (переносное) устройство с аккумуляторной батареей, входящей в набор).

Ключевыми преимуществами комплекса по сравнению с распространенными системами оперативного мониторинга местоположения являются:

- возможности визуализации и анализа пользовательской информации заказчика на входящей в состав комплекса масштабируемой электронной карте;
- возможности хранения данных о перемещении и состоянии контролируемых объектов, подготовки на их основе отчетов, содержащих, в том числе, и визуализированные на электронной карте данные;
- интеграция с программными бизнес-приложениями компании «АНТОР Бизнес Решения» и заказчика.

**Программное решение «ANTOR LogisticsMasterTM»** предназначено для автоматизации работы диспетчеров и позволяет предприятиям, занимающимся доставкой товаров клиентам или транспортировкой грузов на торговые точки и склады, автоматизировать процессы управления перевозками и планирования маршрутов. Данный программный продукт предоставляет возможность не только обрабатывать большое количество информации за короткий промежуток времени, но и четко организовать структуру рабочих процессов, что повышает эффективность работы компании в целом. Основной задачей программного продукта «ANTOR LogisticsMasterTM» является повышение эффективности работы персонала предприятия и автотранспорта за счет автоматизированной подготовки плана доставки продукции. Информационная система помогает диспетчеру сформировать набор рейсов и маршрутов движения, отвечающий следующим требованиям:

- минимальный суммарный пробег всех автомобилей по всем маршрутам;
- максимальная загрузка каждого транспортного средства;
- минимальное использование арендованного транспорта и т.д.

Использование системы «ANTOR LogisticsMasterTM» обеспечивает:

- быстрый расчет эффективного плана доставки, включая расчет загрузки каждого автомобиля и его маршрут;
- сокращение времени планирования рейсов;
- снижение расходов на обслуживание автопарка;
- повышение эффективности использования автотранспорта;
- значительное упрощение задач, стоящих перед диспетчером;
- контроль нецелевого использования автотранспорта;
- создание сопроводительных документов;
- интеграцию с корпоративной системой предприятия.

Дополнительные возможности «ANTOR LogisticsMasterTM» раскрываются при интеграции данного решения с программно-аппаратным комплексом по мониторингу и анализу местоположения и состояния транспорта, грузов и мобильных сотрудников «ANTOR MonitorMasterTM». Комбинированное использование этих программ позволяет сурервайзерам и диспетчерам осуществлять контроль над результатами выполнения планов отгрузок в режиме реального времени, а также решать ряд контрольных и аналитических задач:

- отслеживать текущую информацию о местонахождении своих мобильных сотрудников (транспортных средств) в течение рабочего дня;
- осуществлять ежедневный контроль отклонения фактических параметров использования от запланированных: данная прикладная программа обращает внимание супервайзеров на отклонение от маршрута или графика, если оно выходит за рамки заданной величины (например, «Показать все маршруты с отклонением от плана более чем на 10 %»);

- архивировать полученную информацию для ее дальнейшего использования в целях оптимального планирования приобретения новых автомобилей, использования арендованного транспорта и т.д. [3].

**Механизм планирования маршрутов**

При формировании маршрутов может быть применено несколько стандартных алгоритмов оптимизации, являющихся разновидностями решений классической задачи минимизации пройденного расстояния, известной как «задача коммивояжера». Ее основной принцип заключается в определении замкнутого маршрута, который проходит через каждый пункт обхода один раз и имеет наименьшую длину среди допустимых вариантов. Однако все известные алгоритмы, которые могут быть использованы для получения оптимального решения, требуют полного перебора всех возможных маршрутов. Данная задача весьма сложна и объемна даже для современных высокопроизводительных персональных компьютеров. Поэтому в системах, предназначенных для гражданских целей, применяются эвристические методы, заменяющие исчерпывающий поиск приближенным, что позволяет получить быстрое решение при умеренном проигрыше результата. Существует несколько эвристических методов, осуществляющих приближенный поиск оптимальной топологии маршрута. В таблице приведены названия методов и их алгоритмы построения маршрутов.

| Наименование метода | Алгоритм построения маршрута |
|---|---|
| Метод ближайшего соседа (Nearest Neighbor) | Пункты обхода плана последовательно включаются в маршрут, причем каждый очередной пункт должен быть ближайшим к последнему |
| Метод ближайшего города (Nearest Town) | На каждом шаге алгоритма к текущему множеству пунктов, уже принадлежащих маршруту, добавляется новый пункт, для которого найдется ближайший к любому из них, после чего полученный маршрут заново оптимизируется по выбранному критерию |
| Метод самого дешевого включения (Most Cheap Inclusion) | Похож на предыдущий алгоритм, только включение нового пункта приводит к минимальному увеличению стоимости (длины) маршрута |
| Метод минимального остовного дерева (Minimum Spanning Tree) | Представляет собой три последовательно выполняемых шага. На первом шаге для множества пунктов плана строится кратчайшее остовное дерево (с помощью алгоритма Прима, заключающегося в построении каркаса наименьшего веса графа путем его наращивания за счет присоединения ребра с наименьшим весом, только один конец которого принадлежит фрагменту каркаса). На втором шаге в построенном графе выделяется маршрут минимальной длины, который проходит через каждый пункт не менее одного раза. На третьем шаге из последовательности перемещения исключаются все пункты, повторно вошедшие в маршрут. Полученная топология является искомым приближением решения «задачи коммивояжера» и образует допустимый маршрут. |

Таблица. Эвристических методы и алгоритмы поиска оптимального маршрута

В таблице эвристические методы перечислены в порядке улучшения оценки качества приближенного решения и, соответственно, увеличения вычислительной трудоемкости. Каждый из них обладает своими плюсами и минусами. Наилучшее решение для конкретных исходных данных может быть найдено путем их последовательного при-

менения, а затем выбора того варианта, который отвечает вашим требованиям. В качестве критерия оптимальности может быть выбран пробег, время на маршруте, грузооборот, количество задействованного транспорта (либо их комбинации). В системах планирования подбор оптимального метода чаще всего происходит без участия оператора, который может регулировать данный процесс, лишь оценивая результаты планирования и изменяя критерии оптимальности [4].

## Основные принципы построения систем автоматизированного планирования маршрутов

Рассмотрим основные принципы функционирования перечисленных элементов применительно к решению задачи управления транспортом.

Требования к корпоративным системам со стороны модуля автоматизированного планирования невелики. Современные программы данного класса способны использовать любые базы и источники данных, однако на саму информацию накладываются достаточно жесткие условия. От полноты и корректности данных о товаре и клиенте, поступающих из основной информационной системы в модуль автоматизированного планирования, зависит как качество формируемых маршрутов и рациональность использования ресурсов автотранспорта, так и оптимальность складских бизнес-процессов. Если планирование учитывает, что водители должны преодолеть определенное расстояние до момента пиковой загрузки дорог (это особенно актуально в утренние часы), любая задержка при погрузке может стать причиной срыва уже не одной, а нескольких доставок из разных маршрутов. Следовательно, одной из первоочередных задач компании, решившей автоматизировать процесс планирования, является достижение 100-процентной корректности данных в справочниках номенклатурных позиций [4].

Еще одной серьезной проблемой для автоматизации управления транспортом является формализация информации о клиенте. Любой параметр, касающийся адреса, подъездных путей, специфики разгрузки, наличия лифта при большой этажности здания, требуемого времени доставки, способа расчета (наличный, безналичный), непосредственным образом влияет на успех выполнения доставки. Данное требование зачастую вынуждает производить определенные доработки в корпоративной информационной системе (КИС), что связано с дополнительными расходами, но опыт показывает, что использование геоинформационных технологий (GIS) в системе приема заказов приносит и дополнительные выгоды, например, резко сокращает время заполнения формы заказа, так как основные составляющие адреса выбираются из списков, а дополнительные (район, станция метро, индекс и т.д.) автоматически заполняются на основе базы данных.

GIS дают возможность одновременной работы пользователя с несколькими типами данных: пространственными и атрибутивными. Первый тип данных определяет форму и местоположение объекта и состоит из векторной информации (набор слоев, каждый из которых содержит ряд элементов, как правило, точек, линий и полигонов) и растровой (в виде сплошных изображений: картографическая основа, аэро- или космические снимки). Атрибутивные данные представляют собой дополнительные сведения в виде числовых, символьных и логических параметров (например, площадь, длина, ширина, количество населения, степень загрязнения экологической среды и т.д.), содержащихся в специальных таблицах. Интегрируя широкий набор информации, хранящейся в базах данных, электронных таблицах и разнообразных документах, GIS системы позволяют получить наиболее наглядное представление о ситуации в удобном и легком для понимания формате – электронной карте на экране персонального компьютера.

## Заключение

Рассматривая GIS-системы в контексте автоматизации управления транспортом, можно выделить следующие классы, имеющие одинаковую основу, но отличающиеся функциональностью.

1. Системы планирования маршрутов. В список их основных задач входит:

- отображение электронной карты города в различных масштабах;
- поиск нужного объекта и предоставление о нем справочной информации;
- формирование маршрутов доставки заказов с учетом множества ограничивающих факторов;
- просмотр результатов планирования на электронной карте с возможностью внести ручные изменения;
- подготовка и печать заданий экспедитору (возможно с картой маршрута) и сопроводительных документов;
- проведение анализа и накопления статистики использования транспортных средств.

2. Системы мониторинга и навигации. Помимо функций визуализации и накопления информации, эти программно-аппаратные комплексы за счет интеграции с системами глобального позиционирования предоставляют следующие возможности:

- определение местоположения транспортного средства, оснащенного навигационным оборудованием;
- контроль в режиме реального времени перемещения транспортных средств и мониторинг данной информации на электронной карте (с известной степенью приблизительности);
- возможность сравнения плановой и фактической информации о процессе перемещения транспорта по маршруту;
- принятие обоснованных управленческих решений на основе анализа полученной информации;
- донесение принятых решений до водителя /экспедитора.

Использование систем данного класса позволяет свести к минимуму число задействованного персонала в процессах планирования и диспетчеризации. Они сводятся к настройкам параметров и поддержке в актуальном состоянии справочников, а визуализация с помощью электронной карты исходных данных и результатов планирования значительно облегчает сотрудникам контроль оптимальности сформированного маршрута [4].

При этом, однако, необходимо учесть, что даже самый совершенный алгоритм не способен учесть абсолютно все особенности конкретного клиента, поэтому, основываясь на собственном опыте или руководствуясь оперативно полученной информацией, диспетчеру может потребоваться внести изменения в маршрут. Имея перед глазами в графическом виде полную информацию по конкретной территории обо всех клиентах и задействованном в текущий момент транспорте, он с большей вероятностью примет правильное решение.

## Литература


1. www.autonews.ru/automarket_news/index.shtml?/2005/09/30/1162788
2. www.gazeta.ru/news/auto/2007/02/13/n_1035157.shtml
3. www.antor.ru
4. http://www.transpages.ru/article.php?id_art=36

# РАЗРАБОТКА КОМПЛЕКСА ПРОГРАММ ДЛЯ АВТОМАТИЗИРОВАННОГО ТЕСТИРОВАНИЯ, АНАЛИЗА АРМ ДИЛЕРА МЕЖДУНАРОДНОГО ВАЛЮТНОГО РЫНКА FOREX

**М.С. Тулякова**
**Научный руководитель – к.т.н., доцент Б.А. Крылов**

Проведено исследование различных видов и методов тестирования программного обеспечения, а также изучена предметная область разработки: рынок Forex. Произведена разработка приложения, позволяющего осуществлять стрессовое, нагрузочное, негативное тестирование протокола взаимодействия между сервером и АРМ Диллера. Приложение имеет графический интерфейс и разработано на языке Java 2.0

## Введение

Торговля на международном рынке Forex – одна из составляющих изменения курса мировых валют. Каждый день на рынке Forex продается и покупается с выше пятисот миллионов долларов, евро, фунтов стерлингов и прочих валют. При этом все перемещения денежных масс происходит виртуально, и очень важной частью в этом процессе являются сервера, обеспечивающие возможность совершения сделок, и люди, которые контролируют сделки на рынке (дилера). Соответственно, чтобы сделки проходили честно, быстро и надежно, необходим четкий и постоянный контроль за всеми составляющими серверной платформы, обеспечивающей торговлю на рынке. Одной из них является протокол взаимодействия дилера с сервером, в том числе – определение времени реакции сервера на разные запросы при разной нагрузке, поскольку курс валют меняется до 50 раз в минуту, и сделка, совершенная по курсу, который был упущен из-за того, что сервер не успел отреагировать, может обернуться не только проблемами для клиентов, заключивших сделку, но и отразиться на общемировом курсе валют. Для того чтобы определить узкие места, улучшить производительность и скорость взаимодействия между дилером и серверной платформой, требуется комплекс, который позволил бы совместить в себе нагрузочное, стресс- и функциональное тестирование и был родственен по своей структуре самому серверу.

Одним из составляющих системы является протокол передачи данных между сервером и клиентами, работающий поверх протокола HTTP/HTTPs. Протокол обмена разделен на две части – запросы, передаваемые методом GET, и частично POST. Ответы приходят в XML по протоколу HTTP. В течение долгого времени разработчики наблюдали ошибки, связанные с передачей данных, а также задержки ответов со стороны сервера, но на тот момент не было средств для определения устойчивости протокола и поиска места в коде или протоколе, из-за которого происходят задержки. Именно поэтому потребовалось создать программное обеспечение (ПО), которое позволило бы протестировать и сервер, и протокол, причем с применением разных тестовых подходов – стресс-тестирование, нагрузочное, функциональное и негативное тестирования. Было предложено сделать основной упор на функциональное и стрессовое тестирование, поскольку, в первую очередь, необходимо определить, правильно ли функционирует сервер и как он реагирует на сообщения при повышенной нагрузке. Обычное число пользователей, одновременно подключенных к серверу, в среднем составляет пятьсот человек, во время падения или роста курсов валют это число увеличивается почти вдвое. Поэтому проверка функциональности при стрессовых ситуациях очень важна, потому что с помощью этих тестов и при помощи мониторинговой системы сервера можно определить, в какой части кода происходит задержка сообщений.

## Аналитический обзор существующих систем тестирования

Тестирование – один из важнейших этапов проверки качества разработанного ПО. Основной целью тестирования является увеличение вероятности того, что приложение при любых обстоятельствах будет функционировать надлежащим образом и будет соответствовать установленным требованиям. Тем самым приложение будет удовлетворять ожиданиям конечных пользователей благодаря обнаружению (и последующему устранению) как можно большего числа дефектов [1].

Принято разделять тестирование по уровням задач и объектов на разных стадиях и этапах разработки ПО [3].

(1) тестирование частей ПО (модулей, компонентов) с целью проверки правильности реализации алгоритмов – выполняется разработчиками;
(2) функциональное тестирование подсистем и ПО в целом с целью проверки степени выполнения функциональных требований к ПО – рекомендуется проводить отдельной группой тестировщиков, не подчиненной руководителю разработки;
(3) нагрузочное тестирование (в том числе стрессовое) для выявления характеристик функционирования ПО при изменении нагрузки (интенсивности обращений к нему, наполнения базы данных и т.п.) – для выполнения этой работы требуются высококвалифицированные тестировщики и дорогостоящие средства автоматизации экспериментов.

Каждый из типов применяется на одном или нескольких этапах тестирования ПО. Соответствие типов и этапов тестирования отображено в табл. 1.

| **Вид тестирования** | **Стадия, этап** | **Объект** | **Критерий** |
|---|---|---|---|
| Структурное, надежности | Разработка | Компоненты | Покрытие ветвлений, функции |
| Сборочное | Разработка | Подсистемы | Функциональность, степень проверки компонентов |
| Функциональное | Разработка | Система в целом | Соответствие функциональным требованиям ТЗ |
| Регрессионное | Разработка, сопровождение | Система в целом | Проверка качества внесения изменений |
| Нагрузочное | Разработка, сопровождение | Система в целом | Оценка статистических характеристик системы, соответствие ТЗ, ТТХ, подбор конфигурации оборудования |
| Стрессовое | Разработка, сопровождение | Система в целом | Корректность работы системы при предельных нагрузках |

Таблица 1. Этапы тестирования

## Виды и методы тестирования

*Детерминированное тестирование* (*ДТ*) – это наиболее эффективный метод тестирования, при котором задаются конкретные совокупности исходных данных, которым соответствуют также конкретные значения эталонных результатов. Этот метод позволят не только обнаружить ошибку, но и в ряде случаев локализовать ее.

ДТ основывается на нескольких подходах.

(1) Структурное тестирование или тестирование программы как «белого ящика» (стратегия тестирования, управляемого логикой программы). Подбираются такие тесты,

которые позволяют обеспечить выполнение максимально возможные количества маршрутов, логических ветвлений, циклов и т.п.

(2) Функциональное тестирование или тестирование как «черного ящика» (тестирование по входу-выходу). Абстрагируясь от логики программы, проверяют только входные и выходные функциональные спецификации.

(3) Стахостическое тестирование – это тестирование, при котором исходные тестовые данные задаются множеством случайных величин с соответствующими распределениями, а для сравнения полученных результатов используются также распределения случайных величин. В результате при стохастическом тестировании возможно более широкое варьирование исходных данных, хотя отдельные ошибки могут быть не обнаружены, если они мало искажают средние статистические значения или распределения.

*Модульное тестирование* – это тестирование программы на уровне отдельно взятых модулей, функций или классов. Цель модульного тестирования состоит в выявлении локализованных в модуле ошибок в реализации алгоритмов, а также в определении степени готовности системы к переходу на следующий уровень разработки и тестирования. Модульное тестирование проводится по принципу «белого ящика», т.е. основывается на знании внутренней структуры программы и часто включает те или иные методы анализа покрытия кода.

*Интеграционное тестирование* – это тестирование части системы, состоящей из двух и более модулей. Основная задача интеграционного тестирования – поиск дефектов, связанных с ошибками в реализации и интерпретации интерфейсного взаимодействия между модулями.

*Системное тестирование* охватывает целиком всю систему. Большинство функциональных сбоев должно быть идентифицировано еще на уровне модульных интеграционных тестов. В свою очередь, системное тестирование обычно фокусируется на нефункциональных требованиях – безопасности, производительности, точности, надежности т.п. На этом уровне также тестируются интерфейсы к внешним приложениям, аппаратному обеспечению, операционной среде и т.д. [2]

## Рынок Forex и программные средства для торговли на нем

Форекс (Forex) – от англ. foreign exchange market (международный валютный рынок) сформировался в 1971 году, когда межбанковская торговля после Бретон-Вудского соглашения перешла от фиксированных курсов валют к плавающим. Товаром на международном валютном рынке Forex являются валюты различных стран. Главный принцип на рынке Forex заключается в обмене одной валюты на другую по свободно формирующемуся курсу. При этом курс одной валюты относительно другой определяется равновесие спроса и предложения [5].

Рынок Forex не имеет конкретного места торговли, торговля на валютном рынке осуществляется по телефону или через компьютерные терминалы одновременно в сотнях банков во всем мире. Торговые операции на рынке совершаются круглые сутки в течении всей рабочей недели. Самый распространенный в настоящее время способ торговли на рынке Форекс – торговля через Интернет (интернет-трейдинг). Интернет-трейдинг сделал торговлю на валютном рынке доступной в любой точке мира, при наличии торгового терминала и выхода в Интернет.

Осуществить выход на FOREX возможно только через посредника. Таким посредником может быть дилинговый центр. Эта организация предоставляют канал связи (компьютерный или телефонный) с брокером, который дает котировки валюты и через которого можно совершать торговые операции. Клиент заключает договор с компанией, по которому последняя обязуется по поручению клиента за свой счет и от своего имени осуществлять операции. Главная особенность валютного рынка Forex, привле-

кающая к нему мелких игроков, – это возможность купли и продажи иностранных валют при отсутствии у трейдера всей суммы, необходимой для совершения сделки. Недостающую часть предоставляет дилинговый центр в виде кредитного плеча. Кредитное плечо – это отношение между суммой залога и выделяемым под нее кредитным капиталом. Риск потерь возлагается на клиента, депозит страхует дилинговый центр.

Сервер VTFX – программная платформа, которая обеспечивает бизнес-логику работы клиентов с рынком Forex, а также обеспечивает контроль и безопасность финансовых потоков пользователей и компании. При обращении клиента к серверу первым запросом должен быть запрос на получение TradingSystemInterface (объект, который содержит информацию обо всех доступных пользователю сервисах на сервере. Список доступных сервисов зависит от типа пользователя и сервера). В качестве ответа сервером посылается XML, который содержит информацию о всех сервисах, доступных клиенту для запросов. Следующим запросом к серверу должен быть запрос «login», в ответ на который сервер присылает сообщение с идентификационным номером клиента. Далее клиент получает информацию о данных своего аккаунта и отправляет запросы к серверу на получение предназначенной ему информации. Доставка сообщений проходит по методу pull, при котором клиент сам инициирует получение сообщений. Таким образом, клиент постоянно посылает запросы к определенному сервлету на сервере для получения предназначенных ему сообщений.

Клиентами сервера VT FX являются программы VTDealer и VTTrader. VTTrader – инструмент, с помощью которого трейдер будет торговать валютой, участвуя во всемирной торговле на бирже FOREX. Для использования этой программы необходимо зарегистрировать аккаунт трейдера в компании Visual Trading Systems.

АРМ дилера представляет из себя набор терминалов и программного обеспечения, которое позволяет контролировать все сделки, идущие через платформу VT FX. Это необходимо для того, чтобы отсекать мошеннические сделки, сделки, сделанные по ошибке на большие суммы, а также те, что были совершены по цене, которой на рынке уже нет. Например, клиент покупает 10000 евро за доллары по цене 1.24, но цена ушла в это время до 1.30 за евро. Такое случается, если у клиента Интернет-канал не обладает достаточной пропускной способностью или сервер компании перегружен. Мошеннические сделки появляются, когда кто-либо из пользователей или клиентов компании обнаруживает ошибки в платформе или использует специфические возможности сервера и могут принести большой ущерб компании. Наиболее частый вариант мошенничества – торговля на пипсах.

В общем случае АРМ дилера – это два компьютерных терминала или персональных компьютера, на подключенных мониторах видна вся информация по текущему и прошлому состоянию рынка и сделкам, проводимым клиентами. Также дилер получает всю информацию о котировках сторонних фирм, новостях, которые могут повлиять на рынок и так далее.

### Общая схема работы приложения Leiron

Программа работает как в консольном, так и в графическом режиме. При запуске программы в консольном режиме, необходимо в параметрах запуска указать task-файл с цепочками запросов для сервера. В этом случае программа сразу же начинает посылать запросы к серверу, а результаты писать в консоль. Так же программа пишет лог своей работы в файл access.log.

В графическом режиме у пользователя есть возможность задать task-файл (файл с цепочками команд), init-файл (файл с типами запросов, отрабатываемыми сервером, а также с параметрами этих запросов), лог-файл. Графики времени обработки запросов выводятся только в графическом режиме.

Функциональность приложения Leiron:

- приложение отправляет запросы на сервер согласно заданным в XML-файле цепочкам команд;
- типы запросов, отрабатываемые сервером, а также их параметры можно задать в XML-файле;
- приложение измеряет и записывает в структуру данных время отработки каждого запроса;
- приложение сравнивает ответы сервера с эталонными;
- приложение стоит графики времени выполнения определенной команды или группы команд;

Таким образом, работа программы состоит из нескольких этапов: чтение и преобразование данных из XML-структуры; отправка цепочек запросов серверу; обработка результатов работы программы. Общая схема работы приложения изображена на рис. 1.

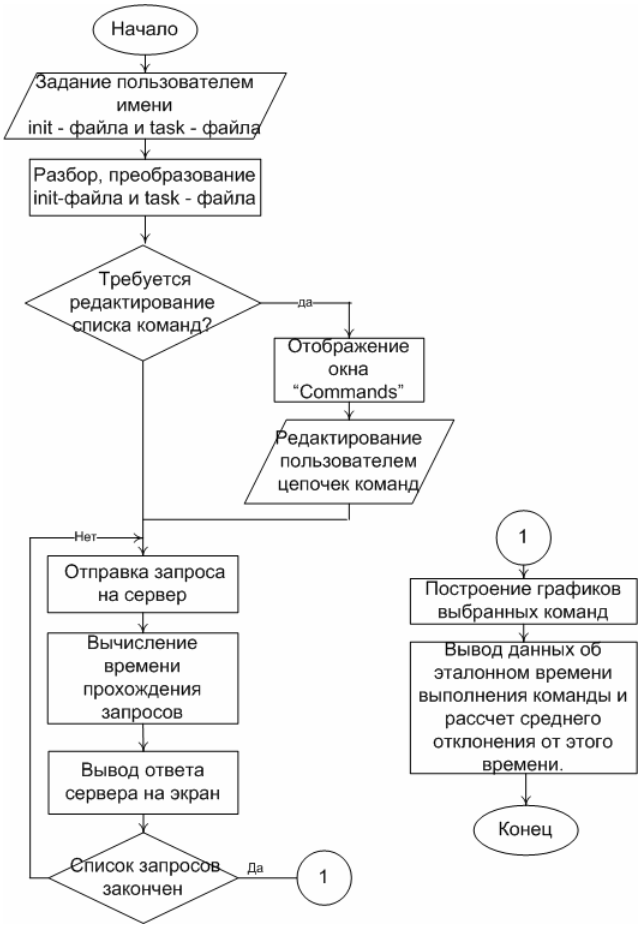



Рис. 1. Общая схема работы программы

**Работа с init-XML файлом и task-XML файлом**

Init-XML файл для каждого запроса, поддерживаемого сервером, содержит информацию об имени запроса, о том, какой сервис сервера обрабатывает данный запрос, типе ответа сервера и о параметрах, необходимых для обработки данного запроса. Каждая цепочка команд, содержащаяся в task-xml файле, должна содержать ip хоста и порт, по которому запрос должен быть отправлен, а каждая команда этой цепочки – параметры, необходимые для выполнения этого запроса. Все команды должны быть уже известны программе, т.е. должны быть заданы в init-файле; параметры также должны соответствовать указанным в init-файле параметрам для данной команды.

Разбор файла происходит при помощи стандартной библиотеки для разбора XML файлов SAX (Simple API for XML). Он основан на последовательной генерации событий, связанных с каждой сущностью при разборе.

### Отправка запросов серверу

При отправке запросов используется структура данных, полученная на предыдущем этапе. Пользователь имеет возможность задавать необходимость выполнения данной группы тестов и количество раз, которое необходимо эту группу выполнить в программе (рис. 2). По умолчанию группа тестов выполняется, а количество повторений этой группы тестов – 1.

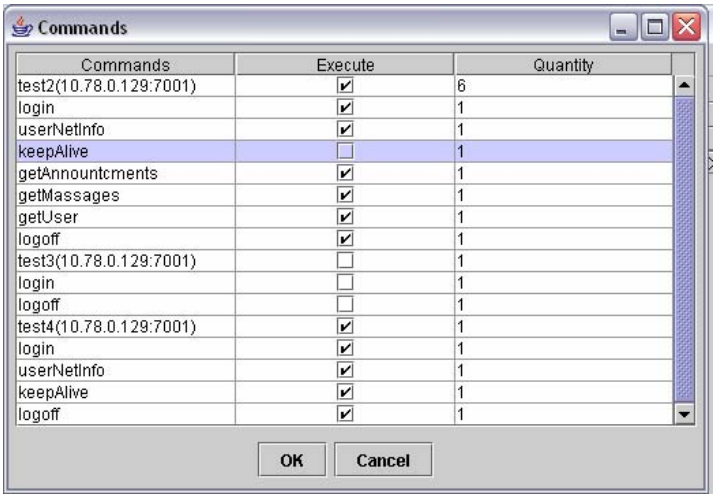



Рис. 2. Окно приложения, позволяющее менять параметры выполнения тестов

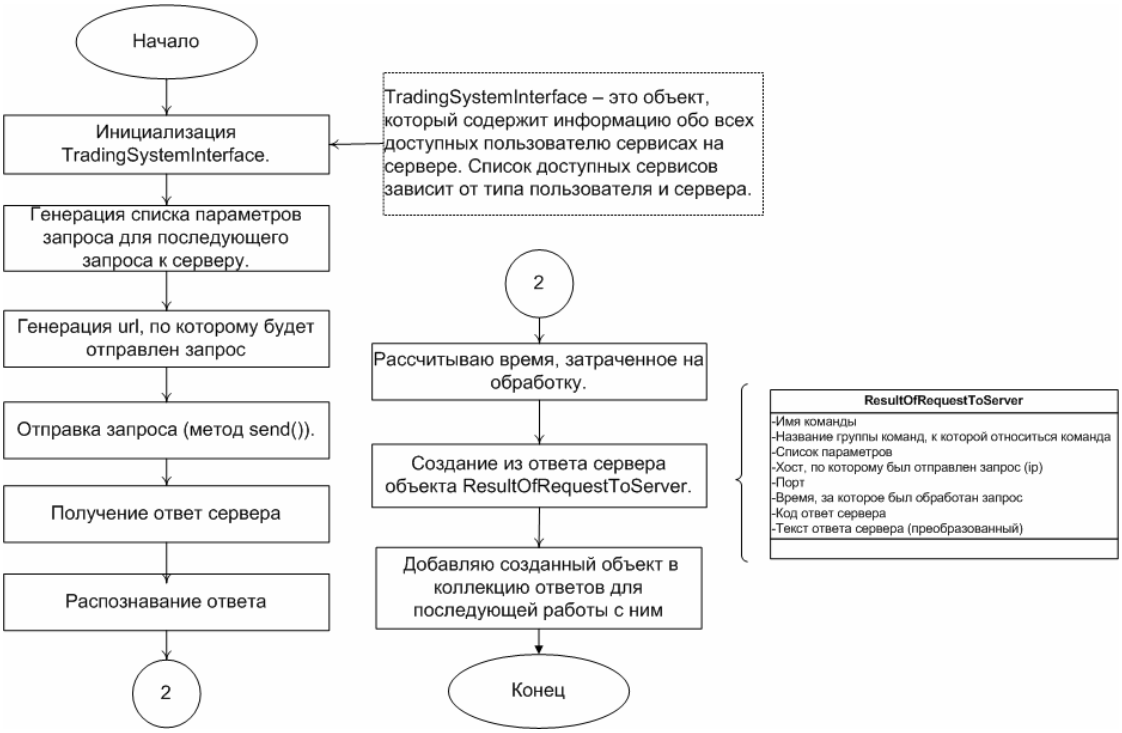



Рис. 3. Структурная схема алгоритма отправки запросов на сервер

При запуске тестирования:

- последовательно перебираются все группы тестов;
- проверяется, нужно ли выполнять текущую группу тестов;

- проверяется, сколько раз нужно повторить текущую группу тестов;
- перебираются все запросы в этой группе тестов;
- проверяется, нужно ли выполнять текущий запрос;
- выполняется текущий запрос.

Структурная схема алгоритма отправки запросов на сервер изображена на рис. 3, а результат работы программы – на рис. 4.

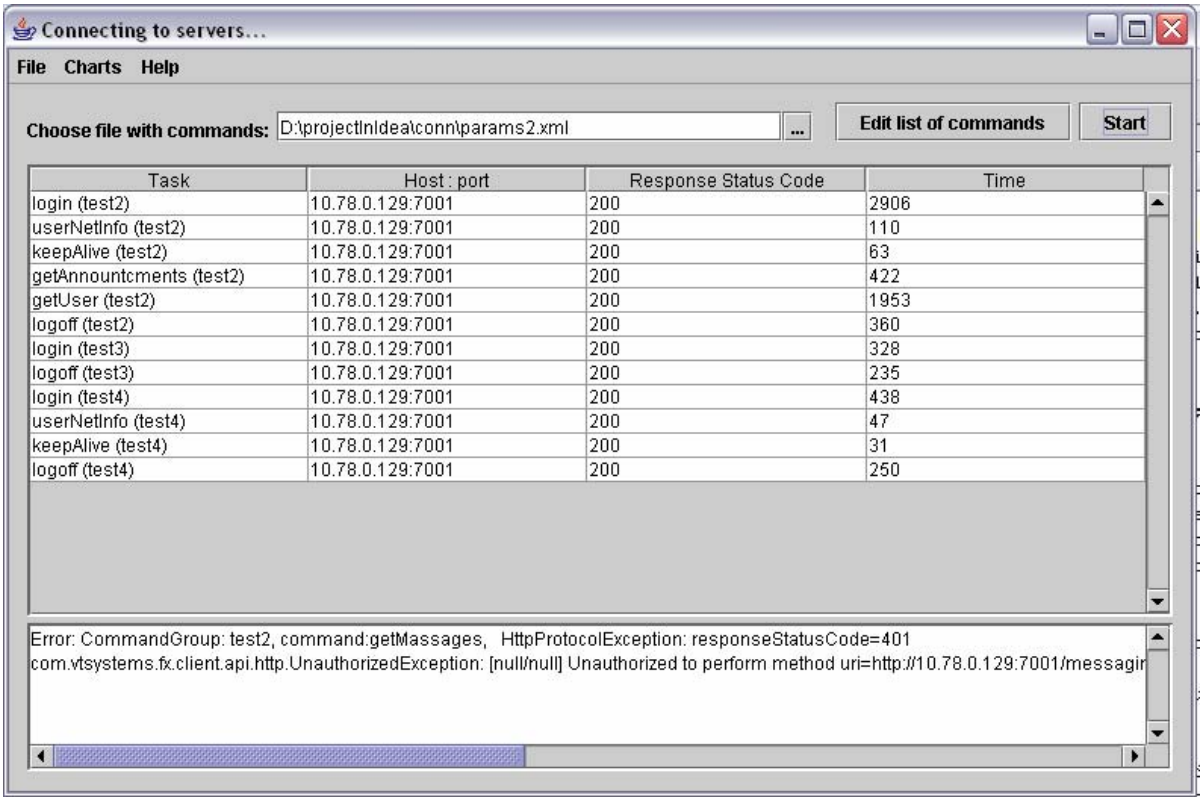



Рис. 4. Главное окно приложения Leiron

### Обработка результатов запросов

Обработка результатов включает в себя построение графиков, показывающих время выполнения очередного запроса для дальнейшего анализа производительности, а также анализ среднего времени выполнения определенного типа запроса и сравнение с эталонным временем для этого типа запроса.

Построение графиков осуществляется с использованием библиотеки JFreeCharts. Программа строит графики двух видов:

- для определенного типа запроса – отображение графика времени, затраченного выполнение всех запросов данного типа. На этом графике также отображается эталонное время выполнения данной команды, которое можно задать в специальном окне. По умолчанию берется среднее время выполнения этой команды;
- для группы тестов – отображение графика времени, затраченного на выполнение каждого отдельного запроса этой группы тестов.

Программа строит график нормального распределения времени обработки для каждого запроса к серверу. Графики, построенные программой, изображены на рис. 5.

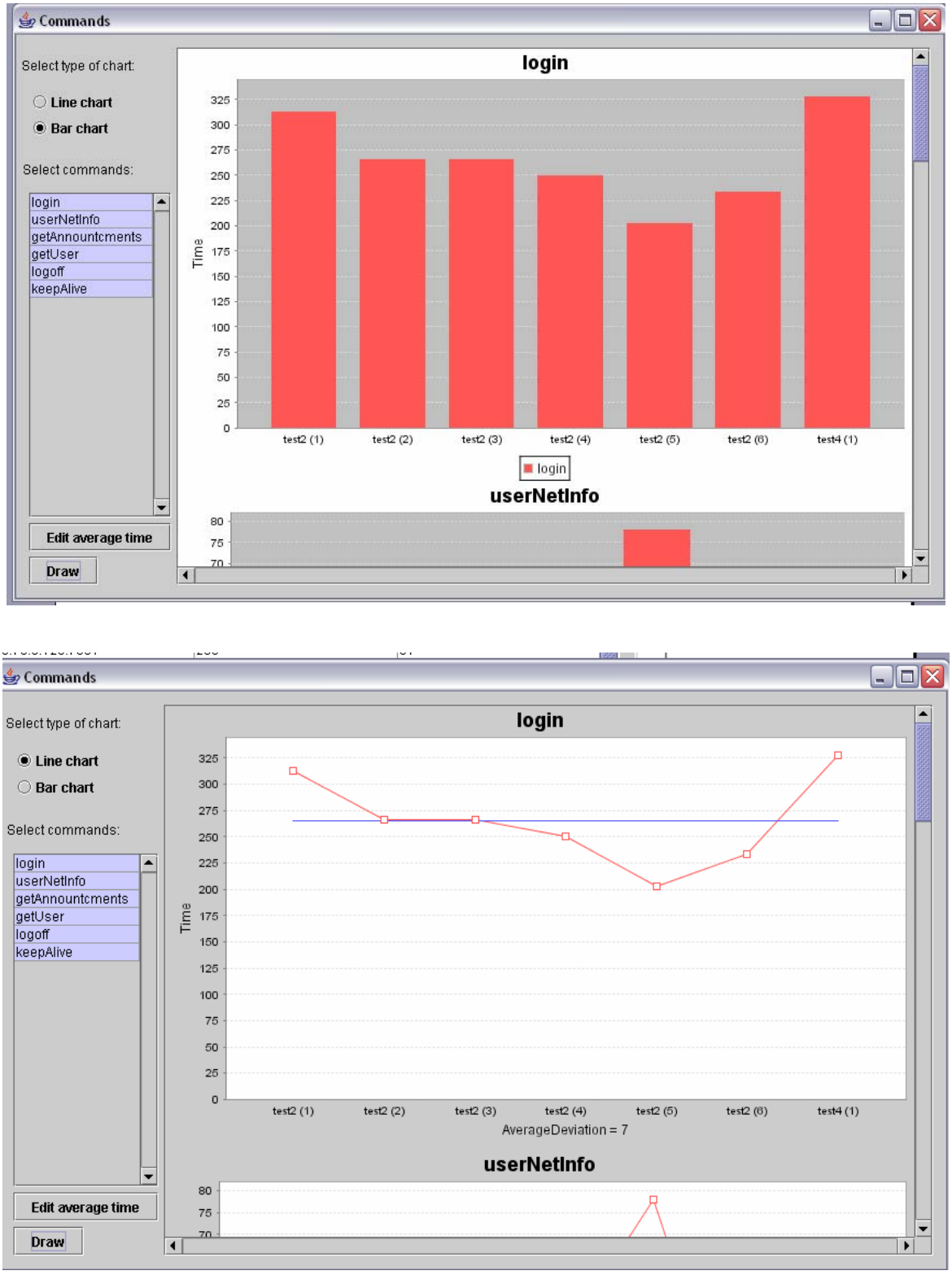



Рис. 5. Окно программы, отображающее графики

## Заключение

Работа выполнена на фирме Визуал Трейдинг Системс и будет использоваться при стрессовом, нагрузочном, негативном и функциональном тестировании протокола взаимодействия VTTrader'а и VTFX. Приложение позволяет не только тестировать, но и анализировать результаты тестов.

В ходе выполнения работы были изучены различные методики тестирования программного обеспечения и выбрана оптимальная методика для решения поставленной задачи. Также было проведено изучение платформы VT FX и схемы ее взаимодействия с сетевыми службами; разработана методика тестирования протокола взаимодействия АРМ дилера и сервера VT FX; выполнена программная реализация разработанных алгоритмов.

**Литература**


1. Макгрегор Дж., Сайкс Д. Тестирование объектно-ориентированного программного обеспечения; К: Диасофт, 2002. 432 с.
2. Канер С., Фолк Дж., Нгуен Енг. Тестирование программного обеспечения, К: Диа-Софт, 2000  544 с.
3. www.software-testing.ru
4. Ведихин А., Петров Г., Шилов Б. Валютные рынки для начинающих и профессионалов, М.: Омега Л, 2005. 428 с.
5. vtsystems.com
6. www.pfgfx.ru/forex

# АВТОМАТИЗАЦИЯ ПРОЦЕССА ПРОГРАММИРОВАНИЯ

**Ю.А. Великоруссов**
**Научный руководитель – к.т.н., доцент Б.А. Крылов**

В работе рассматривается одна из наиболее актуальных и перспективных задач современного этапа развития САПР – автоматизация процесса программирования предметной области изображения. Для решения данного вопроса выбраны методы и инструменты порождающего программирования. Эта технология учитывает преимущества автоматизации применительно к разработке программных средств

## Введение

Сегодня компьютерные науки находятся на пороге кардинального обновления, причем в наибольшей степени это касается области программирования и принципов проектирования. Тенденция к переходу от объектов и вещей к понятиям и характеристикам набрала большие обороты. Именно на этой теории основывается система порождающего программирования; в значительной степени она сформировала такие методики, как инженерия предметной области.

Принципы разработки программного обеспечения, применяемые в настоящее время (включая распространенные объектно-ориентированные методы анализа и проектирования), нацелены на разработку конкретной системы в расчете на определенную задачу и фиксированный контекст. Это методики формирования одиночных систем. Напротив, инженерия предметной области устремлена на создание программных продуктов многократного применения. Она применяется при решении разнообразных задач различных групп заказчиков [1].

Первый международный симпозиум по порождающей и компонентной программной инженерии был проведен в Германии в 1999 году. В нем исследователи попытались свести свои идеи по разработке программного обеспечения воедино, отказавшись от выведения очередных методик путем их противопоставления предшественникам. Объединение новых концепций по многократному применению программных продуктов происходит на многочисленных семинарах и конференциях.

Технологии порождающего программирования способствуют автоматизации процесса программирования предметной области изображения. Они обеспечивают автоматизацию производства промежуточных и конечных продуктов: компонентов и приложений.

## Порождающее программирование

Порождающее программирование – это автоматизированное производство программных продуктов из отдельных компонентов. Переход к автоматическому производству программного обеспечения основывается на выполнении следующих шагов: во-первых, необходимо перейти от разработки одиночных систем к разработке семейств систем – это позволит подготовить «правильные» компоненты реализации; во-вторых, нужно автоматизировать сборку компонентов реализации при помощи генераторов.

Порождающее программирование фокусирует внимание на семействах программных систем, а не на уникальных продуктах. Элементы семейства не строятся с нуля, они генерируют на основе общей порождающей доменной модели, т.е.модели семейства системы компонентов изображения. Она обладает тремя составляющими: средствами определения членов семейства или пространством задачи; компонентами реализации, из которых может быть собран каждый член; и базой знаний о конфигурациях или пространством решений, отображающим спецификацию для члена семейства в конечный продукт [1].

Пространство решений состоит из компонентов реализации во всех возможных комбинациях. Компоненты реализации разрабатываются в расчете на максимальную сочетаемость, минимальную избыточность и предельное увеличение возможностей повторного использования. В пространство задачи входят прикладные понятия и характеристики, посредством которых разработчики прикладного программного обеспечения могут выражать свои потребности. Знания о конфигурациях устанавливают недопустимые сочетания характеристик, настройки по умолчанию, зависимости по умолчанию (расчет некоторых «параметров по умолчанию» может производиться с учетом других характеристик), правила конструирования (некоторые сочетания характеристик превращаются в определенные сочетания компонентов реализации) и правила оптимизации (одни сочетания компонентов реализации могут оказаться лучше других).

Порождающее программирование эффективно применяется для разработки и реализации порождающей доменной модели системы обработки изображений. Основными этапами разработки метода порождающего программирования предметной области изображения являются следующие:

1. моделирование характеристик и понятий предметной области изображения (регистрация изображения, восстановление, улучшение и сжатие изображения, обработка цветного изображения, сегментация, представление и описание, распознавание объектов);
2. проектирование общей архитектуры и выявление компонентов реализации;
3. определение предметно-ориентированных нотаций, при помощи которых будет производиться «заказ» системы;
4. установление знаний о конфигурациях;
5. реализация компонентов реализации;
6. реализация предметно-ориентированных нотаций;
7. реализация знаний о предметной области при помощи генераторов.

Этап моделирования предметной области направлен на разработку характеристических моделей ее основных понятий.

Компоненты системы обработки изображения (подсистема регистрации изображения, специализированные устройства обработки изображения, подсистема отображения, подсистема массовой памяти и выдачи твердой копии, программы для обработки изображения) должны быть совместимы и сочетаемы друг с другом максимальное количество раз. Необходимо минимизировать дублирование кода и максимизировать его повторное использование [2].

Порождающее программирование помогает разработать «правильные» компоненты для системы изображения и после этого создать на их основе автоматическое предоставление лучших компонентов.

## DEMRAL как образец метода инженерии предметной области

Инженерия предметной области изображения – это деятельность по сбору, систематизации и сохранению наработанного опыта создания систем или частей систем в форме средств многократного применения в рамках данной предметной области, а также по обеспечению методов для повторного использования этих средств (поиска, классификации, распространения, адаптации, сборки) в процессе создания новых систем.

Инженерия предметной области применяется при решении разнообразных задач, таких как разработка предметно-ориентированных каркасов и языков, библиотек компонентов и генераторов. Предметно-ориентированные языки в порождающем программировании обеспечивают возможность «заказа» конкретных членов семейства системы формирования и обработки изображения.

DEMRAL (Domain Engineering Method of Reusable Algorithmic Libraries), т.е. «метод разработки алгоритмических библиотек многократного применения на основе ин-

женерии предметной области» – это метод, специализированный для разработки порождающих алгоритмических библиотек [1]. К категории алгоритмических относятся библиотеки численного анализа, контейнеров, обработки изображений, распознавания изображений, распознавания речи, вычислений графов и т.д.

При помощи предметно-ориентированных языков конфигурирования специфицируются конкретные экземпляры понятий – в частности, структуры данных, алгоритмы, объекты и т.д. Ввиду ограниченности вариантов соединения компонентов реализации их правильные конфигурации можно описать на так называемом языке конфигурирования компонентов реализации – ICCL. Предметно-ориентированный язык конфигурирования относится к пространству задачи, а ICCL – к пространству решений. Перевод с одного из этих языков на другой осуществляется через генератор. Предметно-ориентированный язык призван обеспечить пользователю возможность обозначить свои потребности на оптимальном уровне детализации. Задача ICCL, напротив, заключается в том, чтобы добиться максимальной гибкости и возможности повторного использования компонентов реализации (рис. 1).

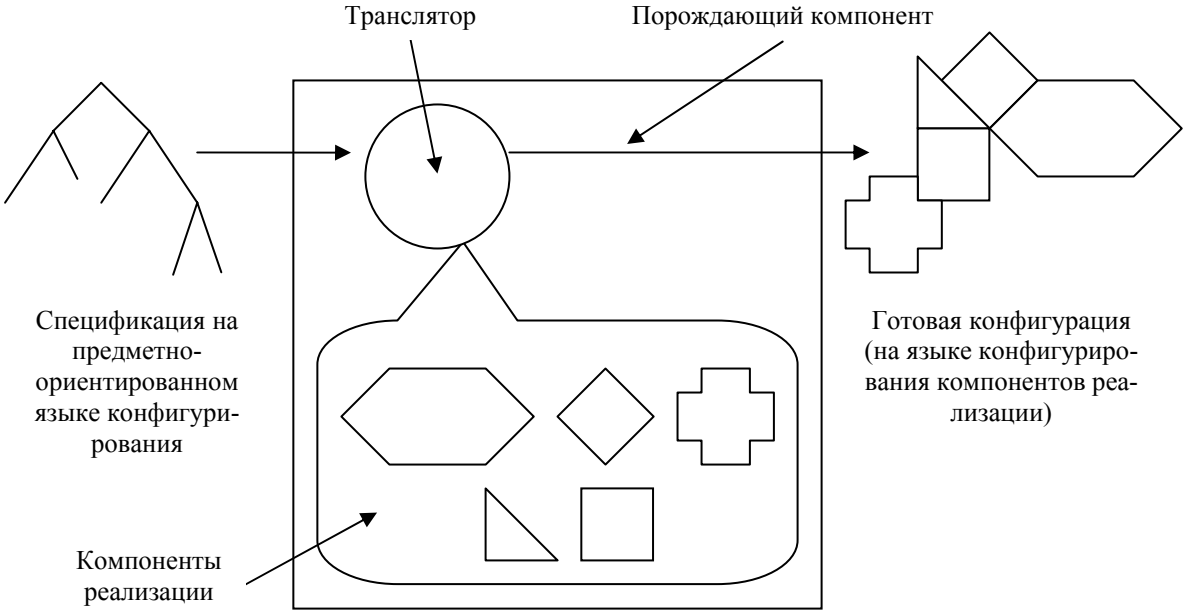



Рис. 1. Реализация предметно-ориентированного языка конфигурирования порождающим компонентом

Благодаря предметно-ориентированным языкам конфигурирования пользователи компонентов имеют возможность указания своих потребностей с оптимальной степенью детализации. Клиентская программа не вводит лишних зависимостей от параметров реализации серверного компонента. Предположим, что клиент хочет запросить у порождающего матричного компонента конкретную матрицу. В ответ на этот запрос матричный компонент должен создать матрицу с некими допустимыми параметрами по умолчанию – к примеру, прямоугольной формы, с действительными значениями элементов, с динамическим количеством строк столбцов и т.д. У клиента должна быть возможность пропускать те или иные характеристики спецификации; в таком случае порождающий компонент определяет эту характеристику как прямое умолчание или вычисляемое умолчание. При обозначении характеристик можно указать некоторые подробности наподобие профиля применения – например, отметить, что нужна плотная или разреженная матрица, матрица, оптимизированная по быстродействию или размещению. У клиента должна быть возможность напрямую указывать те или иные характеристики реализации. Наконец, клиент волен предлагать собственные реализации не-

которых характеристик. Различные уровни детализации при задании конфигурации приводятся на рис. 2.

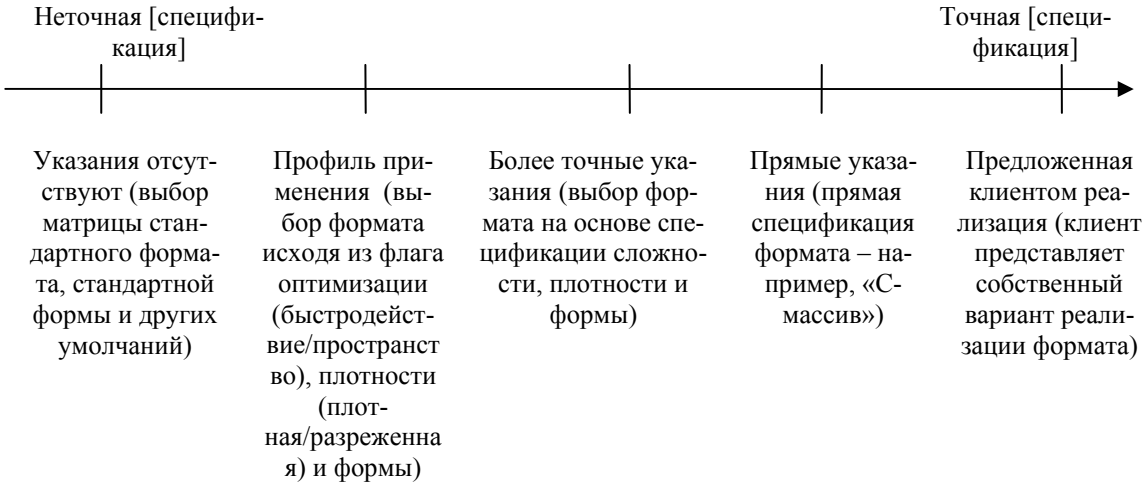



Рис. 2. Детализация при задании изменяемых характеристик

Перечислим отличительные черты алгоритмических библиотек, наиболее часто применяемых для обработки изображения.

1. Основные понятия предметной области в достаточном для поставленных задач объеме фиксируются в виде абстрактных типов данных и оперирующих с ними алгоритмов.
2. Свойства абстрактных типов данных зачастую схожи со свойствами контейнеров – в их числе могут быть матрицы, изображения, графы и т.д.
3. Построения делаются на основе развитых математических теорий – например, на основе линейной алгебры, математических моделей построения изображения, базисных функций, гистограмм изображения [3].
4. Абстрактные типы данных и алгоритмы отличаются многообразием. Например, в большом количестве представлены матрицы, которые различаются по плотности (плотная или разреженная), форме (диагональная, квадратная, симметричная, ленточная), формату хранения и другим параметрам [2].

Для проектирования библиотек DEMRAL обеспечивает достижение следующих целей.

1. Предоставление клиенту высокоуровневого ментального интерфейса библиотеки: клиентский код определяет задачи с точки зрения высокоуровневых понятий предметной области; интерфейс обеспечивает эффективную поддержку многочисленных вариантов понятий; клиентский код определяет задачи с оптимальной степенью детализации (он может «заказывать» реализации понятий по умолчанию или указывать столько деталей, сколько нужно в данной ситуации – не больше и не меньше).
2. Оптимизация времени исполнения и потребления памяти: наличие большого количества вариантов не должно способствовать снижению эффективности; возможности оптимизации должны подвергаться анализу, а те из них, которые будут признаны полезными, – реализовываться; неиспользуемая функциональность должна удаляться.
3. Достижение наиболее высокого качества кода библиотеки: обеспечение максимальной адаптируемости и расширяемости; сведение дублирования и усложнения кода к минимуму.

DEMRAL – это уникальный метод, аккумулирующий концепции инженерии предметной области, генераторов, метапрограммирования, аспектно-ориентированного программирования, объектно-ориентированной разработки программных средств и других областей.

## Генераторы

Современные универсальные языки программирования не предполагают достаточной степени гибкости, ментального, четкого кодирования и высокой производительности. В частности, чтобы повысить производительность кода, приходиться прибегать к его ручной оптимизации, которая разрушает четкую структуру кода. Аналогичное воздействие на код оказывает внедрение аспектов – например, обработки ошибок и синхронизации. Ввиду невозможности замены аспекта альтернативной реализацией страдает гибкость.

Чтобы избежать этих трудностей, работу, связанную с оптимизацией и переплетением, передают генераторам – они вычислят наиболее эффективную реализацию программы обработки изображения на основе высокоуровневой спецификации. Полный цикл работы генератора представлен на рис. 3.

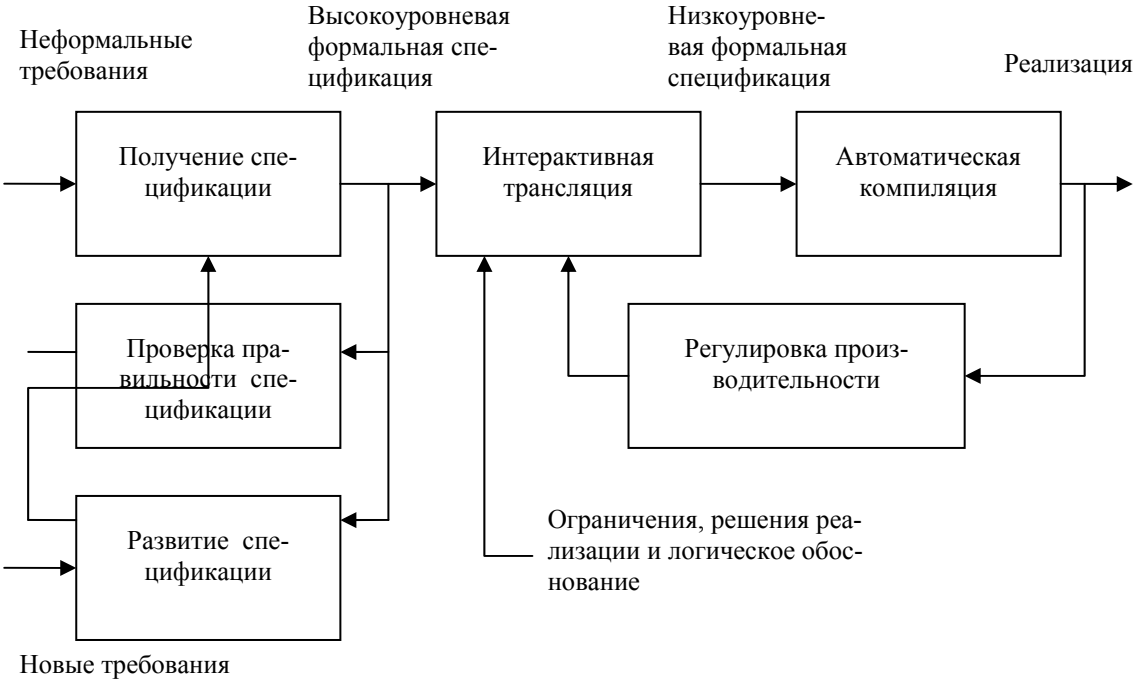



Рис. 3. Модель трансформационного жизненного цикла

Генератор может взять программу, написанную на высокоуровневом языке программирования, и сгенерировать ее реализацию на машинном языке или в виде байт-кода. Генераторы, вырабатывающие реализации моделей (которые довольно часто отображаются в графическом виде) на том или ином языке программирования, присутствуют в большинстве средств автоматизированного проектирования и создания программ обработки изображения. Реализации графически специфицируемых конфигураций компонентов предусматриваются в различного рода компонентных средах и компоновщиках графических пользовательских интерфейсов [1].

В цифровой обработке изображения генераторы выполняют три существенных функции.

1. Повышение ментальности описаний систем. Ментальные описания отличаются тем, что задачи в них излагаются точно и ясно, без лишнего мусора и несущественных деталей реализации. Они аккумулируют все наилучшие качества кода: понятность простота анализа, модифицируемость, удобство сопровождения и т.д. Средством достижения ментальности являются предметно-ориентированные нотации, за реализацию которых и отвечают генераторы.

2. Определение эффективности реализации. Минимальное изменение спецификации может повлечь за собой необходимость в кардинальной переделке реализации.
3. Снятие проблемы масштабируемости библиотек. Чтобы избежать экспоненциального роста объема библиотеки, ее можно разложить на компоненты, соответствующие характеристикам, а затем при помощи вызовов функций или методов скомпоновать. Генераторы обеспечивают сочетание эффективного разложения с высочайшей производительностью.

### Аспектно-ориентированное программирование

Аспектно-ориентированное программирование содержит различные методы и методики разбиения задач на ряд функциональных компонентов, которые выражаются в виде объектов, модулей, процедур и т.д., а также аспектов, которые «пересекают» функциональные компоненты и предусматривают их композицию в целях получения реализаций систем. В настоящее время существует несколько методик, ориентированных на инкапсуляцию различных свойств систем, в том числе и аспектов, которые пересекаются с модульными функциональными элементами.

Фильтры композиции ориентированы на разрешение трудностей, связанных с координацией сообщений в традиционной объектной модели. Объект в рамках модели фильтров композиции изображен на рис. 4.

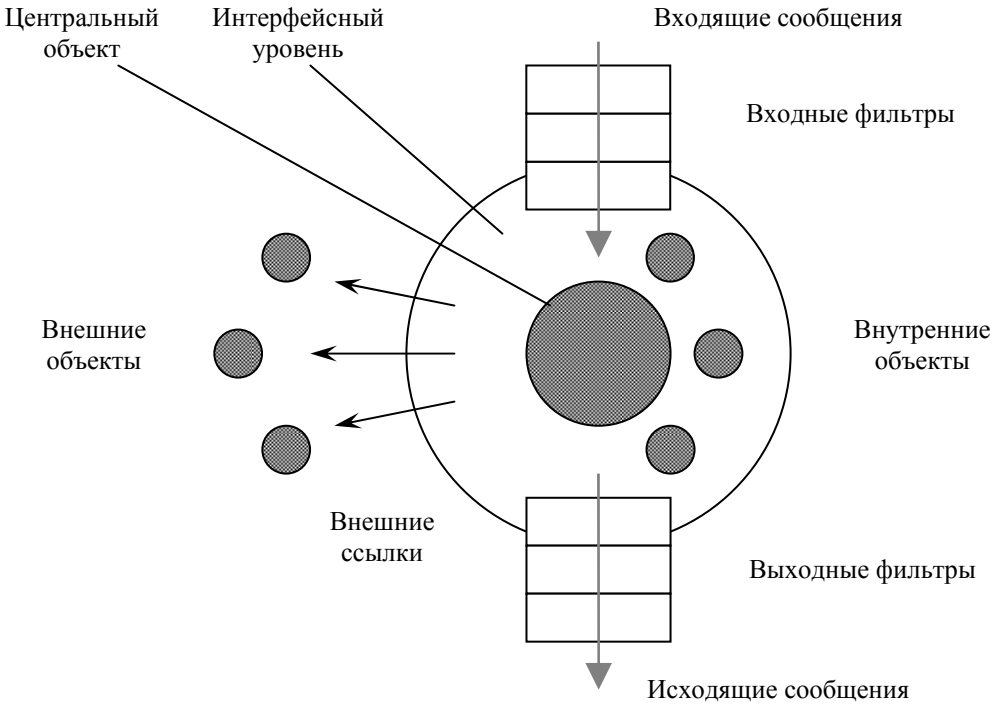



Рис. 4. Элементы объекта в рамках модели фильтров композиции

К примеру, чтобы выразить синхронизацию на интерфейсном уровне объекта, соответствующий код требуется вставить во все подлежащие синхронизации методы. Ручное встраивание кода в методы приводит к смещению функционального и синхронизационного кода и, следовательно, ограничивает возможность многократного применения. Как правило, при расширении класса путем определения его подкласса и его методов приходится обновлять в расчете на этот подкласс всю схему синхронизации и переопределять большинство унаследованных методов, хотя базовые функциональные возможности последних не претерпевают никаких изменений. Причина, по которой код синхронизации приходится обновлять, заключается в том, что он разбросан по различ-

ным методам. Эта и другие подобные проблемы называются аномалиями наследования. Одним из наиболее существенных недостатков традиционной объектной модели является отсутствие механизмов, обеспечивающих отделение функциональной части от кода координации сообщений.

Благодаря фильтрам композиции традиционная объектная модель дополняется рядом различных фильтров сообщений, через которые проходят пересылаемые от объекта к объекту сообщения. Он состоит из интерфейсного уровня и центрального объекта. Центральным объектом может быть любой обычный объект, определяемый в традиционных объектно-ориентированных языках программирования, наподобие Java и C++. Интерфейсный уровень содержит произвольное количество входных и выходных фильтров сообщений. Фильтры могут вносить в проходящие сообщения некоторые изменения – например, менять их имена или переопределять целевые объекты.

Следовательно, одна из их возможных функций заключается в переадресации сообщений другим объектам – внутренним, т.е. существующим в рамках интерфейсного уровня, и внешним, т.е. тем, на которые на интерфейсном уровне установлены ссылки; кроме того, фильтры используются для трансляции сообщений путем замены их имен. Фильтры способны отбрасывать и буферизовать сообщения, порождать исключения. То, какому из упомянутых действий будет отдаваться предпочтение, зависит от типа конкретного фильтра. Существует ряд предопределенных типов фильтров – в частности, фильтры ожидания (для буферизации сообщений), фильтры ошибок (для порождения исключений). Решение о том, будет ли сообщение исправлено или оставлено в первоначальном виде, принимается в зависимости от конкретного сообщения и от условий составления центрального объекта.

Методика фильтрации в обработке изображения крайне эффективна; она позволяет реализовать и обеспечить четкую локализацию синхронизационных ограничений, ограничений реального времени, проверки ошибок по предусловиям и других аспектов. К элементам изображения применяется сглаживающая пространственная фильтрация для расфокусировки изображения и подавления шума, а также фильтры низких высоких частот, частотные фильтры повышения резкости и др. [2].

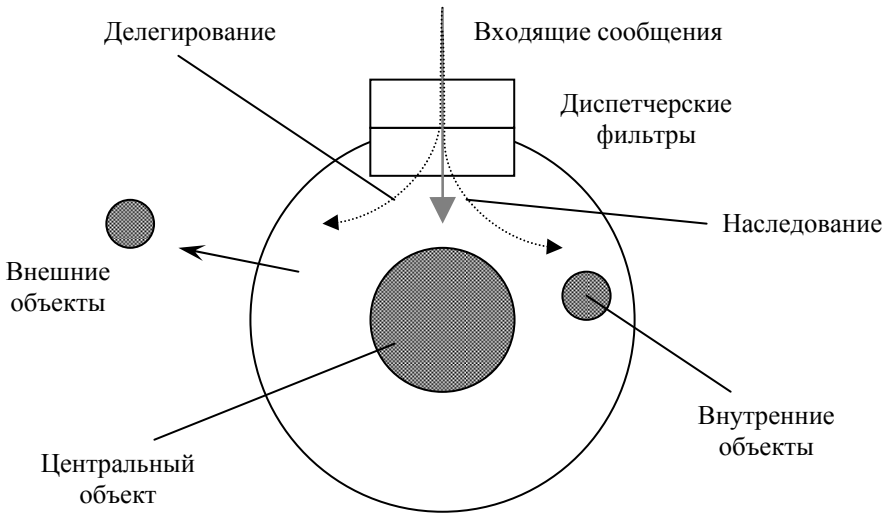



Рис. 5. Делегирование и наследование в рамках модели фильтров композиции

При участии переадресации сообщений реализуются делегирование и динамическое наследование. Делегирование предполагает переадресацию отдельных получаемых делегирующим объектом сообщений другому – уполномоченному – объекту согласно установленной в рамках первого ссылки на последний.

В рамках модели фильтров композиции под делегированием понимается переадресация сообщений внешним объектам и обеспечение отсылки self к исходному получателю. Наследование, напротив, предполагает перенаправление сообщений внутренним объектам, но при этом зарезервированное слово self также должно отсылаться к исходному получателю [1].

На основе тех или иных условий состояния фильтр может делегировать сообщения к различным внутренним объектам. Следовательно, имеет место динамическое наследование, при котором суперкласс объекта способен меняться в зависимости от состояния этого объекта (рис. 5).

## Метапрограммирование

В настоящее время потребность в системах обработки изображения, обеспечивающих возможность быстрой настройки с расчетом на конкретную среду размещения или предполагающих динамическую самонастройку в отношении изменяющейся среды в период прогона, постоянно растет. Необходима возможность корректировки различных аспектов компонентов, конфигурирования и сборки этих компонентов средствами автоматических процессов и анализа работы конечных систем. Некоторые библиотеки способны автоматически адаптировать код, который они поставляют в компиляторную систему. В частности, отбирая алгоритмы и представления данных, наилучшим образом подходящие к целевой платформе, они корректируют свои компоненты.

Автоматическое конфигурирование и сборка, параметризация аспектов, динамическая и статическая приспосабливаемость и адаптивность – чтобы все это можно было провести в жизнь, требуются технологии объявления функциональности компонентов, управления их параметрами и/или реализациями. Этим задачам отвечает метарпограммирование – оно позволяет писать программы, представляющие другие программы и управляющие ими же или сами собой [1].

Приставка «мета» выражает «описание чего-то чем-то» – таким образом, метапрограммы – это программы о программах. Практических примеров метапрограммирования – великое множество: генераторы, компиляторы и интерпретаторы программ, представляющие и управляющие программами на соответствующих языках, а также программы, оснащающие другие программы в целях их тестирования и профилирования, аспектные программы, воздействующие на семантику компонентов, оптимизаторы, средства автоматического рефакторинга.

## Заключение

В данной работе проведен краткий обзор автоматизации процесса развития программных средств возможностями порождающего программирования. Рассмотрены методы анализа и проектирования предметной области изображения, предметно-ориентированные языки, в частности, DEMRAL как образец метода инженерии предметной области, соответствующего задачам порождающего программирования. Также проанализированы некоторые технологии реализации программного продукта, такие как генерация, аспектно-ориентированное программирование и метапрограммирование.

Идея порождающего программирования – строить порождающие модели для семейств систем и генерировать конкретные системы по этим моделям. Для каждого сгенерированного члена многократно используются компоненты реализации и знания о конфигурациях.

Разработка многократно используемых компонентов требует выявление не только общности членов семейств, но и существенных параметров изменчивости. Выявление предметной области и моделирование характеристик предоставляют систематический

способ определения того, какие характеристики и изменяемые параметры нуждаются в немедленной реализации, а какие должны быть запланированы на будущее.

Автоматизация развития станет возможной лишь в случае представления систем при помощи высокоуровневых предметно-ориентированных нотаций, которые позволят явным образом выражать основные свойства систем и фиксировать все проектные решения периода разработки. Чем больше высокоуровневой проектной информации содержится в исходном коде систем, тем шире возможности автоматизации.

Расширяемые среды программирования, стандартизация архитектуры для предметной области и рост рынка компонентов также будут способствовать большей автоматизации и специализации в разработке программного обеспечения.

## Литература


1. Чарнецки К., Айзенекер У. Порождающее программирование: методы, инструменты, применение. СПб: Питер, 2005. 731 с.
2. Гонсалес Р., Вудс Р. Цифровая обработка изображения. М.: Техносфера, 2006. 1072 с.
3. Ritter G., Joseph N. Wilson. Handbook of Computer Vision Algorithms in Image Algebra / CRC Press, Boca Raton. 1997. P. 437–442.

# 2 МИКРОЭЛЕКТРОНИКА, ДЕФЕКТОСКОПИЯ И ДЕФЕКТООБРАЗОВАНИЕ В ПРОЦЕССАХ ПРОИЗВОДСТВА И ЭКСПЛУАТАЦИИ ЭЛЕМЕНТНОЙ БАЗЫ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ И СИСТЕМ УПРАВЛЕНИЯ

## ЩЕЛОЧНОЕ ВСКРЫТИЕ МАКРОПОР ПРИ ИЗГОТОВЛЕНИИ КРЕМНИЕВЫХ СТРУКТУР СО СКВОЗНЫМИ КАНАЛАМИ

**Г.В. Федулова**
**Научный руководитель – А.А. Нечитайлов (ФТИ им. А.Ф. Иоффе РАН)**

В работе рассматривается одна из важных задач при получении кремниевых структур на основе макропористого кремния – задача вскрытия пор с целью получения сквозных каналов в кремниевой пластине. Для решения данного вопроса использовался метод анизотропного щелочного травления.

### Введение

Кремниевые макропористые структуры со сквозными каналами используются в разных областях науки и техники – например, в оптике в качестве противорассеивающих сеток и оптических фильтров, в микромеханике в качестве микронасосов и фильтров частиц. Одно из интенсивно развивающихся направлений последних лет – использование макропористых кремниевых структур в качестве электродов портативных топливных элементов (ПТЭ) и микроканальных реакторов.

Одной из важных задач при получении кремниевых структур со сквозными макропорами – электродов для ПТЭ на основе макропористого кремния [1] – является вскрытие пор с целью получения сквозных каналов в кремниевой пластине. Обычно это достигается путем шлифования обратной стороны пластины до появления сквозных пор по всей поверхности пористой части образца. Однако такой метод при своей простоте имеет ряд недостатков – забивание пор частичками шлифующего материала, возможность получать только плоские структуры, необходимость получения толстых пористых слоев для придания приемлемой механической прочности образцам. Одним из альтернативных путей вскрытия макропор является стравливание кремниевой подложки в результате щелочного травления в растворе едкого кали при защите пористой части и других не подлежащих утоньшению частей оксидной маской $SiO_2$. Такой подход обеспечивает возможность получения тонких пористых слоев и структур более сложной геометрической формы.

### Постановка задачи

Для практической реализации данного подхода необходимо выполнение ряда условий. Нужно уметь точно определять границу подложки с пористым слоем, чтобы не произошло растворение макропор. Необходимо также подобрать условия проведения технологических процессов формирования оксидной маски достаточной толщины на пористой структуре и режимов последующего щелочного травления. Это нетривиальные задачи, так как макропористый кремний – материал со свойствами, отличающимися от монокристаллического кремния. Прежде всего, это материал с развитой поверхностью. Это определяет его относительно большую реакционную способность. Кроме того, при окислении макропористых слоев в результате разности мольных объемов

кремния и его оксида в них возникают существенные механические напряжения [2], приводящие зачастую к разрушению структуры.

Целью работы явилась разработка метода щелочного вскрытия макропор для получения кремниевых пластин со сквозными каналами.

## Экспериментальная часть

В качестве исходных структур использовали пластины кремния типа КЭФ-15, размером 30×30 мм, ориентированные в плоскости (100), толщиной 350–400 мкм, содержащие макропористую область, сформированную методом фотоэлектрохимического анодировании, в виде круга диаметром 20 мм с глубиной пор 200–250 мкм. Использовали два типа образцов: с порами, полученными посредством предварительной фотолитографии и с самоорганизованной структурой. Во всех случаях поры имели средний диаметр 3 мкм с периодом (расстоянием между порами) 8 мкм.

Перед формированием термической оксидной маски образцы отмывали по стандартному технологическому процессу:

1. отмывка в диметилформамиде;
2. отмывка в перекисно-аммиачном растворе;
3. отмывка в плавиковой кислоте;
4. отмывка в кислотно-перекисном растворе.

После каждой операции отмывки производилась промывка в деионизованной проточной воде. По окончании всего процесса отмывки осуществляли сушку пластин в центрифуге. Формирование термической оксидной маски $SiO_2$ проводили стандартным путем в атмосфере влажного воздуха при 1050 °С. Вскрытие окна в оксидной маске на подложке для последующего щелочного травления проводили в специальной ячейке в растворе HF, разбавленной 1:1.

Щелочное стравливание подложки проводили в термостатированном 44 % водном растворе KOH при температуре 70 °С и перемешивании раствора (рис. 1). При этом использовали два варианта – с открытым образцом и с дополнительной изоляцией образца со стороны пористой части от раствора с помощью вакуумной резины.

Контроль степени вскрытия и качества пор проводили с помощью микроскопических исследований.

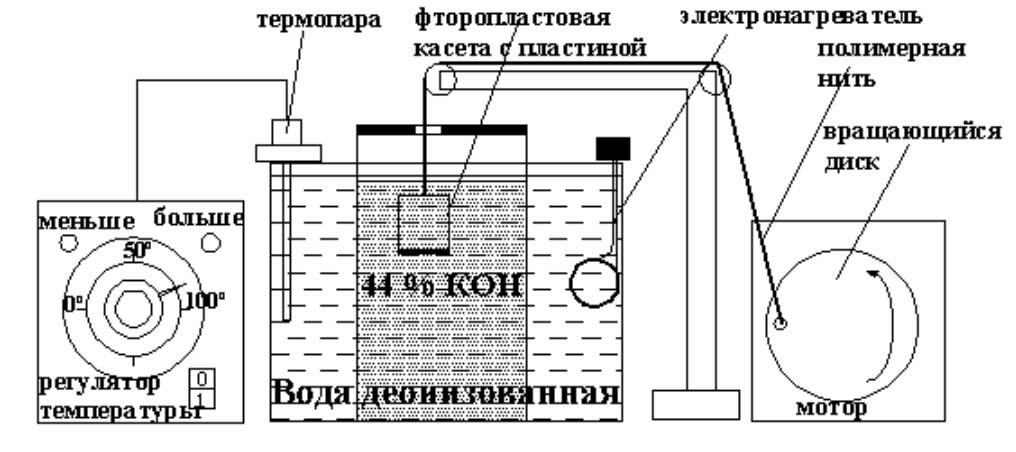



Рис. 1. Схема установки щелочного анизотропного травления кремния

## Обсуждение полученных результатов

Подбор толщины оксидной маски в случае макропор явился отдельной задачей. Формирование термической оксидной маски на поверхности макропор связано с появ-

лением существенных механических напряжений, приводящих к деформации платины. При этом оказалось, что при последующем снятии оксидной маски в случае относительно небольших напряжений происходит восстановление первоначальной формы образца. Однако, когда толщина оксида достигает некоей критической величины, происходит необратимая деформация, и образец остается изогнутым даже после снятия оксида кремния с поверхности макропор. Это – нежелательное явление, и с целью минимизации деформаций была подобрана минимально необходимая толщина оксидной маски. При этом авторами установлено, что процесс щелочного растворения маски $SiO_2$ на пористой части (при последующем щелочном вскрытии) происходит неравномерно и быстрее, чем на гладкой поверхности.

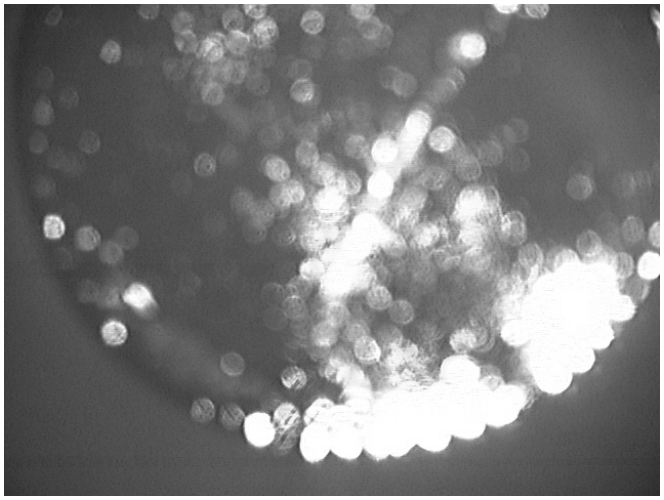


Рис. 2. Микрофотография (на просвет) пористой части после щелочного вскрытия без дополнительной защиты пор. Светлые области соответствуют дыркам в структуре

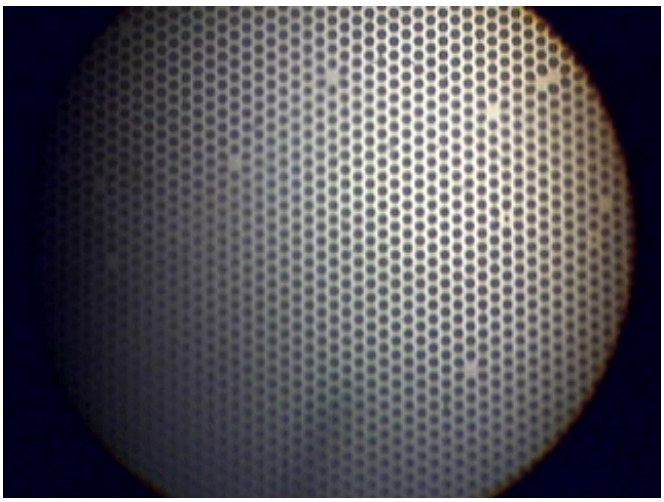


Рис. 3. Микрофотография пористой части после щелочного вскрытия с дополнительной защитой пор резиновой прокладкой

В ряде областей на пористой части маскирующие свойства $SiO_2$ оказались весьма слабыми, что привело к быстрому местному растворению оксидной маски. Авторами сделано предположение о том, что на неоднородность устойчивости разных областей маски к щелочи влияют именно механические напряжения. Можно предположить, что наиболее напряженные области (например, по периметру пористой части, где изгиб наиболее сильный) растворяются быстрее. Действительно, как видно на рис. 2, области у края макропористой части разрушены больше.

С целью защиты пористого слоя от раствора щелочи при щелочном вскрытии он был дополнительно закрыт резиновой прокладкой, что оказалось весьма эффективным. Как видно на рис. 3, дефектные области отсутствуют. При этом для стравливания кремниевой подложки толщиной до 200 мкм достаточная толщина оксидной маски $SiO_2$ составила 0.65–0.7 мкм.

После получения окисленных пластин необходимо было сделать окно в оксиде кремния, через которое будет происходить щелочное анизотропное травление кремния до вскрытия пор. Для этого необходимо было определить точную границу пористой области с обратной стороны образца. Благодаря изгибу пластины после окисления эта граница очень четко определялась визуально без дополнительных устройств.

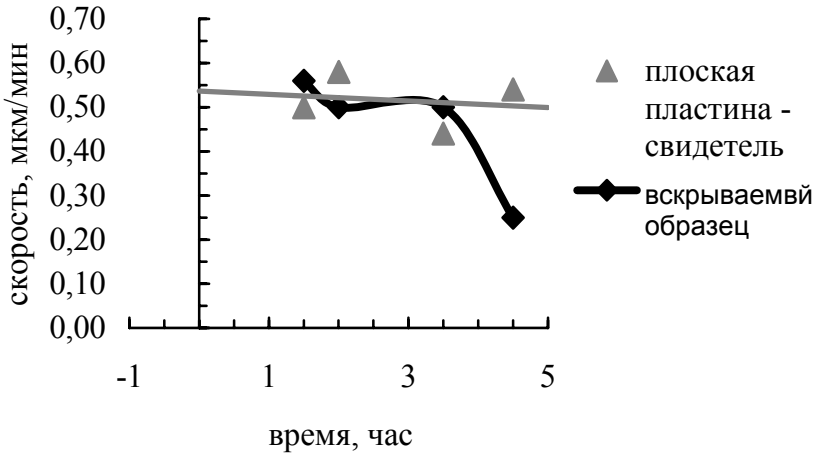



Рис. 4. Зависимость скорости травления при щелочном вскрытии макропор

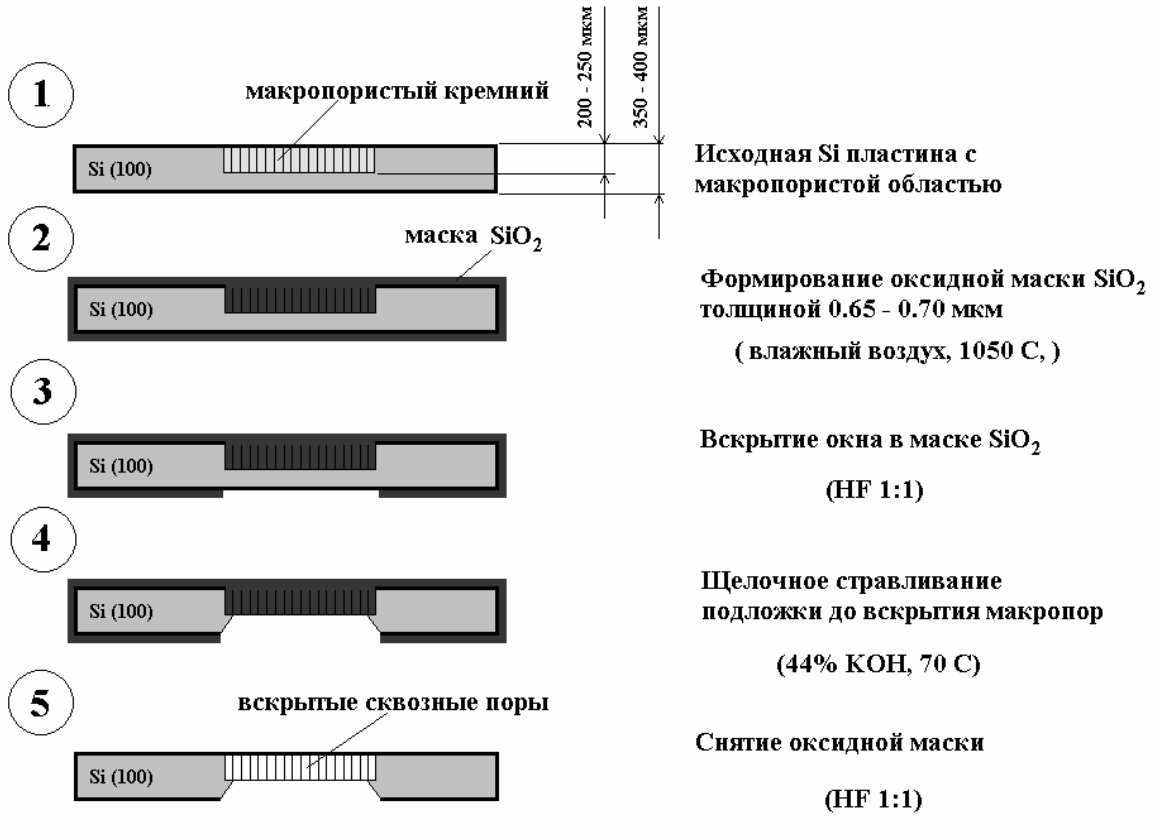



Рис. 5. Технологическая цепочка щелочного вскрытия макропор

Следующей важной задачей явилась необходимость детектирования границы подложки с пористым слоем. Для этого была измерена зависимость скорости щелочного травления от времени для вскрываемого образца и плоской пластины – свидетеля (рис. 4). На основании полученных данных вычисляли необходимое время щелочного травления. Уменьшение скорости травления во времени для вскрываемого образца, в отличие от плоской пластины, можно объяснить сильным влиянием гидродинамического режима и диффузионной составляющей в растворе. При углублении травящейся структуры ухудшается конвекция раствора, появляются застойные зоны. При этом происходит уменьшение скорости травления.

На рис. 5 схематически показана технологическая цепочка щелочного вскрытия макропор. Суммарное время процесса составляет 6–8 часов в зависимости от толщины подложки. В результате такого вскрытия были получены кремниевые структуры со сквозными каналами практически по всей площади пористой части с ровным фронтом щелочного травления.

## Заключение

В работе проведено детальное исследование процесса щелочного вскрытия макропор на образцах макропористого кремния на основе КЭФ-15 ориентации (100) с большой площадью пористой части (круг площадью ~ 3 см$^2$) при получении электродов для ПТЭ. Установлены закономерности изменения во времени скорости стравливания подложки при фиксированных температуре и концентрации щелочи. Скорость травления уменьшается. Показано, что формирование оксидной маски толщиной 0.65–0.7 мкм необходимо и достаточно для стравливания подложки толщиной 150–200 мкм, а щелочное травление необходимо проводить при дополнительном изолировании пористой части от щелочного раствора, например, с помощью вакуумной резины. Разработана технологическая цепочка щелочного вскрытия макропор, включающая следующие основные этапы.

1. Предварительное формирование на всей поверхности образца оксидной маски толщиной 0.65–0.7 мкм путем влажного термического окисления кремния.
2. Вскрытие в маске окна травления со стороны подложки посредством растворения оксида кремния в плавиковой кислоте.
3. Последующее щелочное стравливание подложки до пористого слоя в термостатированном (70 °С) 44 % растворе KOH в специальной ячейке с перемешиванием раствора и с дополнительной защитой пластины со стороны пористого слоя вакуумной резиной. Длительность процесса контролируют по времени. Средняя скорость травления ~ 0.5 мкм/мин.
4. Снятие оксида кремния со всей поверхности образца в растворе плавиковой кислоты.
5. Микроскопический контроль качества вскрытия пор.

Методом щелочного вскрытия получены структуры макропористого кремния со сквозными каналами с толщиной пористого слоя 200–250 мкм, находящегося в центре пластины толщиной 350–400 мкм.

## Литература


1. Астрова Е.В., Бобыль А.В., Горячев Д.Н. и др. Кремниевые технологии для водородной энергетики. / Тезисы докладов Международного Форума «Водородные технологии для производства энергии». М., 2006. С. 188–190.
2. Астрова А.А., Ратников В.В., Ременюк А.Д., Шульпина И.Л. Исследование деформаций и дефектов кристаллической решетки, возникающих при окислении макропористого кремния. // ФТП. 2002. Т. 36. Вып. 9. С. 1111–1121.

# АНАЛИЗ СОВРЕМЕННОЙ ПАТЕНТНОЙ ЛИТЕРАТУРЫ ПО СИЛЬФОННЫМ ЭЛЕМЕНТАМ ДАТЧИКОВ СИСТЕМ УПРАВЛЕНИЯ

**А.Ю. Буданова, В.А. Крылов, О.И. Пирожникова**
**Научный руководитель – д.т.н., профессор В.Л. Ткалич**

Проведенный анализ отечественных и зарубежных работ по сильфонным элементам позволил выделить четыре основных направления исследований по их модернизации. Выявлена высокая актуальность дальнейшего развития сильфонных чувствительных узлов датчиков систем управления. В работе оценены перспективы развития сильфонных элементов, аргументирована перспективность исследования сильфонов для широкого круга задач, определены тенденции их развития.

## Введение

Сильфон – это манометрический упругий элемент, широко используемый в различных областях техники. Он представляет собой тонкостенную, гофрированную в окружном направлении трубку, способную давать значительные перемещения под действием давления, осевой или поперечной силы и изгибающего момента. При осесимметричном нагружении сильфона его характеристика близка к линейной, а эффективная площадь практически постоянна [1, 2].

Сильфоны по типу изготовления делятся на бесшовные и сварные. Последние могут быть сконструированы таким образом, что они способны выдерживать большие перегрузки давлением. Данный фактор обусловливает их широкое применение в соответствующих областях [3]. Сильфоны используются в качестве чувствительных элементов приборов для преобразования в перемещение или в усилие различных измеряемых параметров: давления, температуры, уровня расхода и т.д.

Сильфоны могут развивать значительные перестановочные усилия, что обеспечивает малый порог чувствительности приборов и позволяет использовать их в качестве элементов силовых приводов. Они также служат в различных приборах компенсаторами теплового расширения жидкости, что объясняется их высокой податливостью и способностью значительно изменять объем. Возможность получения сильфонов, обладающих малой осевой и изгибной жесткостью, позволяет успешно применять их в качестве разделителей сред, а также упругих выводов осевых и угловых перемещений. Сильфоны широко применяются и как компенсаторы теплового расширения трубопроводов, элементы гидравлических дистанционных передач. В последнем случае используется свойство сильфонов значительно изменять объем [5].

Основными рабочими характеристиками сильфонов, как и других упругих элементов, являются те, которые определяют его способность деформироваться под действием нагрузки. К ним относятся упругая характеристика, жесткость и чувствительность, а также точность для тех сильфонов, которые используются в качестве чувствительных элементов измерительных приборов [5, 6].

Целью работы явилось проведение патентного поиска и анализа по основным усовершенствованиям сильфонов за период с 1972 по 2006 годы и выявление основных направлений исследований по модернизации данного упругого элемента датчиков систем управления. Проведенный анализ позволил выделить четыре основных направления усовершенствований сильфонных элементов.

## Расширение области применения сильфонов

С целью расширения области применения сильфон выполнен в виде замкнутой камеры, имеющей в поперечном сечении прямоугольный профиль, две параллельные стенки которой выполнены плоскими, а две другие снабжены прямолинейными гофрами.

С целью расширения области применения сильфонов между оболочками, их поверхностями образованы геометрические полости, заполненные сжимаемой средой, причем шаги гофров оболочек кратны друг другу. Данное изобретение обеспечивает сильфонам постоянство их жесткости при изменении температуры материала, что позволяет их применять в условиях осевых перемещений, пульсаций давления и изменения температуры в пневмогидросистемах.

С целью расширения области применения сильфонов, за счет получения нелинейной характеристики усилия и перемещения, пружины выполнены из проволоки с монотонно уменьшающимся диаметром, причем больший диаметр проволоке равен ширине впадины гофра. Данное изобретение позволяет использовать сильфоны в конструкциях упругих измерительных преобразователей (датчиков), и может найти применение в приборах для измерения параметров жидкости, пара или газа в зависимости от значения давления контролируемой среды.

С целью расширения области применения сильфонный узел снабжен размещенной внутри сильфона направляющей втулкой с двумя продольными пазами в ее нижней части, причем наружная поверхность втулки введена в контакт с одной парой роликов. Данное изобретение позволяет использовать сильфоны в арматуре в узлах с двумя и более сильфонами, к которым предъявляются требования высокой вибрационной и ударной стойкости.

С целью расширения области применения полиобъемный сильфон содержит не менее двух расположенных одна в другой гофрированных оболочек. Гофры оболочек имеют одинаковый шаг, опираются друг на друга по аксиально расположенным впадинам, а оболочки сварены друг с другом со стороны торцов. В межсильфонной полости, образованной смежными оболочками, создают избыточное давление.

С целью расширения области применения полиобъемный сильфон содержит не менее двух расположенных одна в другой гофрированных оболочек одинакового шага, но различного диаметра гребней гофр, сваренных со стороны торцов друг с другом, и отличается тем, что указанные оболочки опираются друг на друга по аксиально расположенным впадинам. Кроме этого в межсильфонной полости, образованной смежными оболочками, создается избыточное давление.

С целью расширения области применения отверстия мембран, составляющих сильфон, не являются центральными, а смещены к наружному контуру мембран. Это позволяет получить сильфон, у которого отверстие не является центральным и сквозным, а имеет форму, необходимую для дополнительной очистки газов. Смещение внутреннего отверстия в мембранах сильфона может быть различным и зависит от конкретного случая их применения.

С целью расширения области применения в мембранном сильфоне, содержащем изогнутые кольцевые мембраны, их края попарно и попеременно жестко соединены. Соединение выполнено совместным изгибом стенок соседних мембран на $180^{o}$ дважды в противоположных направлениях.

### Повышение прочностных характеристик сильфонов и надежности

С целью повышения работоспособности волны, расположенные на боковой поверхности гофра сильфона, выполнены по кривой затухающих колебаний от впадины гофра к его вершине.

С целью повышения надежности работы сильфона путем уменьшения внутренних напряжений вершины и впадины его гофр расположены по прямолинейным образующим двух однополостных гиперболоидов вращения, с разными углами наклона этих образующих.

С целью повышения надежности работы сильфона увеличивают его устойчивость путем утолщения одной из боковых поверхностей гофр. Данное изобретение позволяет использовать сильфон в компенсирующих и уплотнительных устройствах.

С целью повышения ресурса работы сильфона его выполняют из плоских колец из полимерного материала, соединенных друг с другом швом по внутреннему и внешнему диаметру. Кольца выполняются двухслойными, а пространство между ними заполнено раствором полимера или способным к полимеризации при контакте с воздухом мономером. Данное изобретение увеличивает эластичность и увеличивает работоспособность, снижая остаточную деформацию. Это позволяет использовать сильфоны для защиты шарнирных и телескопических соединений от воздействия внешней среды.

С целью повышения ресурса работы сильфона исключаются сварные соединения между слоями армирующих колец, при этом каждое кольцо выполнено из свернутой в спираль ленты переменной ширины, а внешний виток спирали выполнен замкнутым.

С целью повышения надежности и нагрузочной способности сильфона, его выполняют из эластичного материала с одной утолщенной стенкой гофра. Утолщения выполняются в гребнях гофров, а во впадинах установлены армирующие кольца с V-образной формой поперечного сечения, вершина которого обращена к оси сильфона, причем наружный диаметр армирующих колец больше наружного диаметра гофров.

С целью улучшения эксплуатационных характеристик путем увеличения податливости и ресурса, первая из каждой пары сваренных гофр установлена впадиной внутрь сильфона, вторая – впадиной наружу, а концы последней изогнуты на 90 градусов и вставлены первую гофру. Данное изобретение позволяет использовать сильфоны для герметичного подвижного соединения двух полых деталей прямоугольного сечения.

С целью повышения работоспособности сильфона его корпус выполнен из узких гофр и, по меньшей мере, двух широких гофр, стенки которых сопряжены между собой выпуклыми и вогнутыми поверхностями. Выпуклые и вогнутые поверхности в продольном сечении выполнены в виде дуги окружности. Отношение высоты $H$ между выпуклой поверхностью широкого гофра и сопряженной с ней вогнутой поверхностью узкого гофра к высоте $h$ между выпуклой поверхностью и вогнутой поверхностью узкого гофра составляет от 0,5 до 0,8. Отношение радиуса $R$ выпуклой поверхности широкого гофра к радиусу $r$ выпуклой поверхности узкой гофры составляет от 2,6 до 5,0, при этом высота $H$ широкого гофра меньше радиуса $R$. Широкие гофры расположены по краям сильфона.

С целью повышения ресурса работы и надежности сильфонов в сварном сильфоне, содержащем последовательно соединенные с возможностью сближения и раздвижения профилированные концентрическими кольцевыми участками мембраны, крайние из них жестко соединены с концевыми деталями, а каждая из остальных соединена с одной из соседних мембран внешним, а с другой – внутренним сварным швом, каждая пара мембран в зоне соединения их внутренним швом выполнена с круговым ребром жесткости в виде пояска, отогнутого на расстоянии не менее 0,5 мм от указанного шва под углом не менее 15° к поперечной плоскости сильфона. Благодаря наличию ребра жесткости и геометрии мембран область сварного шва полностью или частично разгружена от изгибающих механических нагрузок, и разрушение сварного шва в процессе эксплуатации не происходит.

С целью повышения ресурса работы и надежности сильфонов снижаются внутренние напряжения в гибком элементе. Это обусловлено тем, что армированный сильфон содержит гофрированную трубу и усиливающие элементы: опорные полукольца, чехлы-полукольца, полукольца средние, кольца ограничительные, кольца компенсаторные, торцевые шайбы и втулки. При подаче рабочей среды во внутреннюю полость гофрированной трубы ее гофры расширяются совместно с чехлами-полукольцами.

С целью повышения прочности и надежности работы сильфонов производится защита герметичного скрепления от действия деформаций и нагрузок, возникающих от действия массы конструкции. Это обеспечивается за счет того, что мембранный сильфон состоит из мембран крайних и мембран средних, герметично скрепленных попарно и попеременно по наружным и внутренним контурам, а также скрепленных жестким соединением попарно и попеременно, выполненным совместным изгибом на $180^{o}$ два раза в противоположных направлениях стенок соседних мембран. Жесткое соединение выполнено со стороны оси сильфона.

**Улучшение функциональных характеристик сильфонов**

С целью повышения чувствительности упругого элемента стенки гофра сильфона выполняются разной длины. Данное изобретение позволяет использовать сильфоны в измерительных, регулирующих и предохранительных устройствах.

С целью повышения рабочего хода мембрану сильфона выполняют содержащей кольцевые гофры, в поперечном сечении прямолинейные участки, сопряженные криволинейными участками гофр. Причем прямолинейные участки гофр имеют длину, по меньшей мере, в два раза превышающую радиус кривизны криволинейных участков гофр. Данное изобретение позволяет использовать меньшие усилия для деформации сильфона.

С целью повышения функциональных характеристик сильфона выпуклые и вогнутые поверхности выполнены в продольном сечении корпуса в виде дуги окружности. Отношение высоты $H$ между выпуклой поверхностью широкого гофра и сопряженной с ней вогнутой поверхностью узкого гофра к высоте $h$ между выпуклой поверхностью и вогнутой поверхностью узкого гофра составляет 0,5–0,8. Отношение радиуса $R$ выпуклой поверхности широкого гофра к радиусу $r$ выпуклой поверхности узкой гофры составляет 2,6–5,0, при этом высота $H$ широкого гофра меньше радиуса $R$. Кроме того, широкие гофры расположены по краям сильфона.

С целью повышения функциональности сильфона его промежуточный слой выполнен из биметаллов с зеркальным расположением его слоев (активного и пассивного) по отношению к каждому полугофру сильфона (выступу и впадине гофра). Расположение биметаллов по периметру окружности сильфона может быть кольцевым либо дискретно дуговым, последнее может иметь угловое смещение от полугофра к полугофру либо в пределах групп полугофров или спиральную расположенность в упомянутой цилиндрической трубе и, как следствие, в сильфоне. Кроме того, все изложенное может иметь еще и дискретную расположенность в осевом направлении от полугофра к полугофру или в пределах групп полугофров, а свободные от биметаллов объемы промежуточного слоя гофрированной части сильфона, переходной зоны и концевой цилиндрической части заполнены материалом наружного и внутреннего слоев сильфона или аналогичным ему.

С целью повышения функциональных характеристик сильфонов многослойный сильфон имеет внутренний непроницаемый металлический слой и наружные перфорированные слои. Последние чередуются со слоями, выполненными из металлической сетки, что устраняет замкнутые полости в слоях сильфона при сохранении его прочности и гибкости.

С целью повышения гибкости и прочности сильфонов устраняются замкнутые полости между слоями сильфона. Это обеспечивается наличием пористых сетчатых (перфорированных) оболочек с наружной поверхности монолитной оболочки сильфона. Кроме того, сильфон содержит концевые цилиндрические участки и гофрированный средний участок. Гофрированный участок сильфона имеет внутреннюю монолитную металлическую оболочку и несколько наружных оболочек, изготовленных из металли-

ческой сетки (один вариант), из перфорированных металлических оболочек (второй вариант) и слоев сетки и перфорированных металлических оболочек.

### Повышение работоспособности сильфонов при внешних воздействиях

С целью уменьшения габаритов сильфона в сжатом состоянии и повышения его работоспособности при высоких внешних давлениях, одна половина каждого гофра выполнена виде хлопающей мембраны с большим прогибом, а вторая – в виде жесткого тарельчатого элемента с опорной поверхностью, симметричной поверхности мембраны.

С целью повышения вибрационной и ударной стойкости сильфонного узла к воздействию вибраций и ударов применяется равномерное распределение хода штока между сильфонами независимо от их жесткости за счет того, что промежуточная втулка жестко закреплена с сепаратором.

С целью повышения работоспособности сильфонов в условиях воздействия вибрационных и ударных нагрузок устраняются соударения колец сильфона. Для этого в сильфоне, состоящем из закрепленных на кольцах мембран, на внутренних поверхностях колец равномерно по окружности расположены оси, на которых установлены рычаги, связанные между собой и образующие цепь шарнирных параллелограммов.

С целью повышения ресурса работы сильфонов при высоких внешних давлениях и в условиях воздействия вибрационных нагрузок полости между оболочками заполнены под давлением жидкостью, причем в полости, образованной первой и второй от оси сильфона оболочками, давление меньше, чем внутреннее рабочее давление, а в каждой последующей полости давление меньше, чем в предыдущей.

С целью повышения вибрационной и ударной стойкости сильфонного узла, его функциональных характеристик, а также упрощения конструкции, в сильфонном узле с последовательным соединением сильфонов промежуточной втулкой, содержащем шток и контактирующие с ним тела качения, последние выполнены в виде роликов. В промежуточной втулке выполнены попарно соосные отверстия, в которых с возможностью вращения установлены торцевые части роликов.

### Заключение

В результате анализа отечественных и зарубежных работ по сильфонным элементам и проведенного патентно-информационного поиска выявлена высокая актуальность дальнейшего развития сильфонных чувствительных узлов датчиков систем управления.

Проведенный патентный поиск и последующий анализ усовершенствований сильфоных элементов за период с 1972 по 2006 годы позволил выделить четыре основных направления исследований:

1. расширение области применения сильфонов;
2. повышение прочностных характеристик сильфонов и надежности;
3. улучшение функциональных характеристик сильфонов;
4. повышение работоспособности сильфонов при внешних воздействиях.

В работе аргументирована перспективность исследования сильфонных чувствительных элементов для широкого круга задач:

- в качестве манометрических чувствительных элементов;
- в пневматической регулирующей аппаратуре;
- в качестве чувствительного элемента приборов для преобразования давления в перемещение;
- в качестве элементов силовых приводов;
- в качестве компенсаторов теплового расширения жидкостей;
- в качестве разделителей сред – фильтров;

- в качестве упругих выводов осевых и угловых перемещений;
- в строительстве трубопроводов, машиностроении, самолетостроении и вакуумной технике.

Анализ приборов, сконструированных на базе сильфонных элементов, за период с 1960-х годов по настоящее время позволил оценить перспективы развития данных элементов, что наглядно отражено на рис. 1 в координатах «временные промежутки»– «количество изобретений на базе сильфонов».

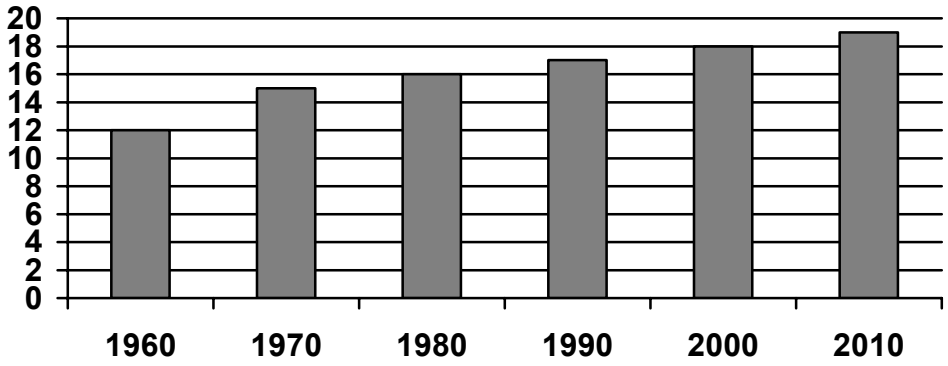



Рис. 1. Перспективы развития сильфонных элементов

Высокий интерес к сильфонным элементам подтверждается большим количеством работ, посвященных теории, экспериментальным исследованиям и методам расчета сильфонов.

Использование ЭВМ открыло новые возможности в разработке методов исследования сильфонов. Появились работы с более точными решениями задач определения жесткости и напряжений в сильфонах. Кроме того, появилась возможность формулирования и решения новых задач, которые прежде считались недоступными инженерам из-за их сложности. К ним относятся задачи расчета сильфонных элементов сложной формы (например, сильфонов с переменной толщиной гофров), а так же задачи, связанные с учетом малых нелинейных эффектов поведения сильфонов [5, 6].

Из всего вышесказанного можно заключить, что сильфоны широко применяются в различных областях техники, причем спектр их применения постоянно расширяется. Данное обстоятельство обусловливает высокий интерес науки к исследованию сильфонов и определяет тенденции развития данных манометрических элементов.

## Литература

1. Андреева Л.Е. Упругие элементы приборов. Изд. 2. М.: Машиностроение, 1981. 392 с.
2. Пономарев С.Д., Андреева Л.Е. Расчет упругих элементов машин и приборов. М.: Машиностроение, 1980. 326 с.
3. Жибарева И.Н. О проектировании упругих чувствительных элементов (стандартные измерительные сильфоны). // Приборы и системы управления. 1998. № 11. С. 43–50.
4. Осипов С.В. Разработка методов расчета нестабильности характеристик упругих элементов сильфонного и мембранного типа, 1987.
5. Ткалич В.Л. Исследование форм эластик упругих чувствительных элементов (УДК 62.27). // Научное приборостроение. 1999. Т. 9. №2. С. 53–58.
6. Ткалич В.Л., Степанова Н.Е., Момзикова Т.Н. Анализ современных методов и алгоритмов решения уравнений динамики. // Деп. ВИНИТИ 28.06.00, № 1813 – В00. 9 с.

# ИССЛЕДОВАНИЕ КАЧЕСТВА ФОТОЛИТОГРАФИИ В СЛОЯХ ПОЛИКРИСТАЛЛИЧЕСКОГО КРЕМНИЯ ПРИ ФОРМИРОВАНИИ ЗАТВОРОВ КМОП ИС

**А.С. Бабков, Н.В. Лопатнёва**
**Научный руководитель – д.т.н., профессор А.М. Скворцов**

В работе приведено исследование распределения длины поликремниевого затвора МОП транзисторов по площади пластин с готовыми КМОП ИС. Рассмотрено влияние различных методов травления, а также степени лигирования пленки поликремния на изменение геометрии затвора.

## Введение

Фотолитография – важнейший технологический процесс при формировании ИС, существенно влияющий на процент выхода годных микросхем. При изготовлении КМОП ИС 590 серии процесс фотолитографии применяется 11 раз, что, естественно, во многом определяет выход годных микросхем в серийном производстве. Одним из таких процессов является формирование геометрии поликремниевого затвора n-канальных и p-канальных МОП транзисторов указанной серии.

При формировании затворов МОП-транзисторов используются пленки $SiO_2$ и $Si_3N_4$. В нашем случае для подзатворного окисла микросхем 590 серии используются пленки $SiO_2$, толщина которых составляет 1200 Å. Окисление подложек производят либо в сухом, либо в увлажненном кислороде. Достоинствами сухого окисления являются изначально низкий встроенный заряд и меньшая пористость. Недостаток заключается в том, что при таком окислении требуются высокая температура (от 1050 °С и выше) а также высокое качество отмывки пластин. Для улучшения электрофизических свойств границы раздела структуры $Si\text{-}SiO_2$ в окислитель добавляют HCl, но от этого оборудование быстро корродирует, снижается радиационная стойкость затворной композиции.

К достоинствам термического окисления во влажном кислороде можно отнести то, что здесь используются более низкие температуры (от 800 до 1000 °С), а также получаются более радиационно-стойкие затворные композиции. Недостатками метода являются увеличенные по сравнению с сухим окислением встроенный заряд и пористость.

После формирования на поверхности подложки затворного окисла и осаждения на окисел пленки поликристаллического кремния (ПК) проводится процесс фотолитографии, в результате которого формируются затворы МОП транзисторов и первый слой коммутации. Для травления ПК используется следующий состав травителя в объемных процентах: азотная кислота $HNO_3$ – 50 %, дионизованная вода – 49,7 %, плавиковая кислота HF – 0,3 %.

В настоящей работе исследовалась зависимость геометрии поликремниевого затвора МОП-транзистора от различных способов травления ПК и степени его легирования.

## Экспериментальная часть

На первом этапе работы ставилась задача исследовать распределение длин затворов n-канальных и p-канальных МОП транзисторов на кремниевых подложках с готовыми, разбракованными по функционированию КМОП ИС 590 серии, и установить корреляцию между длиной затвора и работоспособностью микросхем. На втором этапе требовалось установить причины растрава затворов МОП-транзисторов и выработать рекомендации по улучшению качества травления поликремния.

На рис. 1 приведена фотография подложки с готовыми микросхемами. Кристаллы с забракованными микросхемами помечены маркером. Как видно из рис. 1, большая часть годных микросхем расположена в центре пластины. На каждой пластине измеря-

лась длина поликремниевых затворов n-канальных и p-канальных МОП транзисторов в каждом третьем кристалле в ряду во всех рядах.

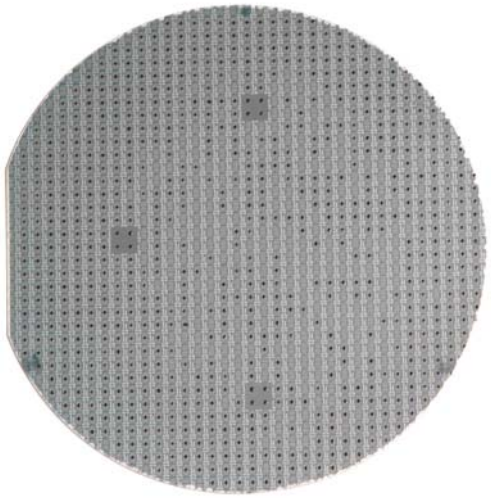


Рис. 1. Фотография пластины с готовыми микросхемами
после 100 % контроля на функционирование

Измерение затворов проводилось на 3-х пластинах, отобранных из разных партий, для получения более достоверных результатов. Результаты измерений заносились в таблицы, а затем строились распределения размеров затворов отдельно по каждой пластине.

На рис. 2 приведена схема сечения участка кристалла с годной микросхемой, на котором располагается два взаимодополняющих МОП-транзистора в изолированных областях («карманах»). Левый «карман» легирован на большую глубину акцепторной примесью для формирования в нем n-канального МОП-транзистора. В правом «кармане» сохраняется исходный кремний, имеющий электронную проводимость. Длина измеряемых затворов обозначена на рис. 2 буквой *l*.

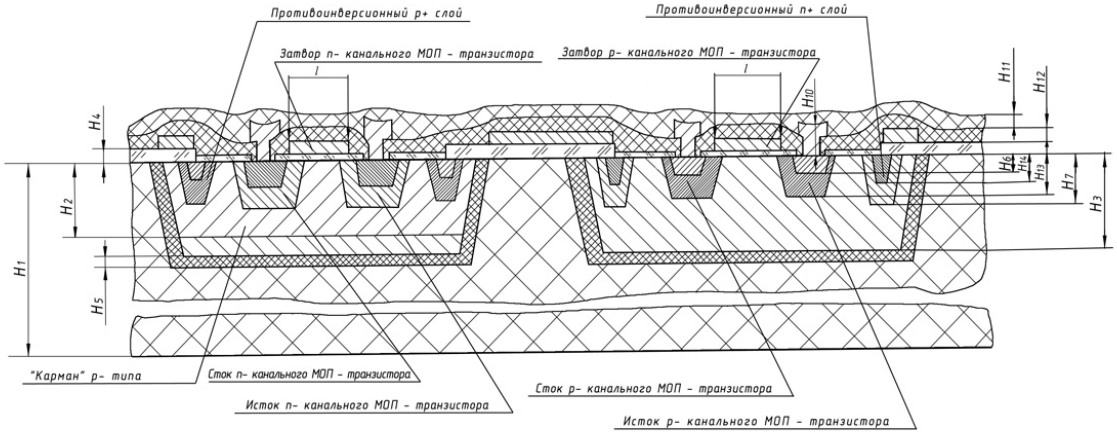



Рис. 2. Сечение двух взаимодополняющих МОП-транзисторов

Из результатов измерений следует вывод, что длина затвора имеет разброс от 10 до 12,5 мкм; меньшие размеры – по краям пластины, а максимальные размеры – в центре пластин. Длина затвора влияет на выход годных микросхем: у годных микросхем длина затвора составляет не менее 11,25 мкм. Диаграмма распределения размеров каналов на одной из пластин в относительных единицах приведена на рис. 3. Соответственно, мож-

но сделать вывод, что процент выхода годных микросхем в определенной степени коррелирует с геометрией затвора, т.е. зависит от качества фотолитографии в пленке ПК.

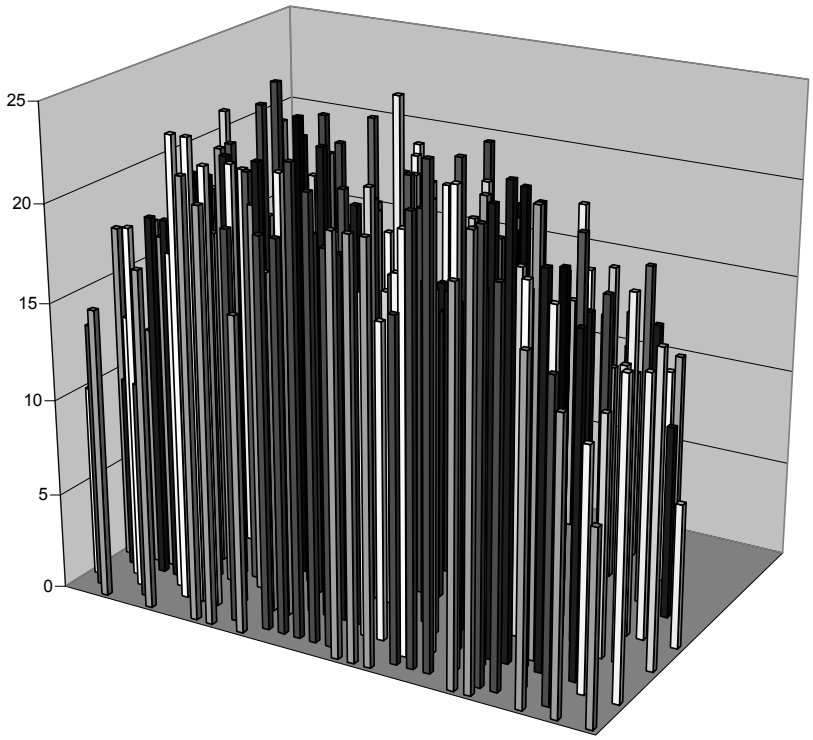



Рис. 3. Распределение длины затвора по площади пластины

Для установления причин, приводящих к растраву поликремниевых затворов, было специально изготовлено 5 кремниевых пластин со слоем поликремния на окисле. На поверхности поликремния по существующей технологии была сформирована защитная маска из фоторезиста. Перед травлением ПК были произведены замеры длины затворов в маске фоторезиста на каждой пластине через 3 ряда с шагом 8–10 затворов в каждом ряду, из которых было видно, что еще до травления имеется небольшой разброс размеров (в пределах 0,1–0,2 мкм).

Три пластины из пяти подвергались стандартному процессу травления, т.е. использовалось травление ПК методом погружения в травитель. Разница в подготовке образцов заключалась в следующем.

- Пластина № 1 подвергалась обработке строго в соответствии с технологией, т.е. перед фотолитографией производилось легирование пленки ПК фосфором.
- Пластина № 2 отличалась от первой тем, что обратная сторона пластины была защищена задубленной пленкой фоторезиста.
- У пластины № 3 легирование пленки ПК фосфором не производилось.

После проведения стандартного травления были отмечены следующие особенности. На первых двух пластинах растрав затворов был одинаковым, и величина растрава соответствовала ранее полученным результатам, хотя наблюдалась меньшая скорость травления ПК на пластине № 2. Скорость травления пленки ПК на пластине № 3 была существенно меньше, чем на первых двух, однако распределение растрава затворов по площади пластины был таким же, как и у первых двух пластин.

Пластины № 4 и № 5 готовились по стандартной технологии (см. пластина № 1). На этих пластинах был изменен метод травления. При травлении ПК на пластине № 4 использовалось перемешивание травителя в ванне в процессе травления, а травление на пластине № 5 осуществлялось путем полива травителя на рабочую поверхность.

В результате было обнаружено, что в обоих случаях существенно возрастает скорость травления пленки ПК. Измерение размеров затворов показало как уменьшение

самого растрава затворов, так и уменьшение разброса размеров по площади пластин. Однако по абсолютным значениям отклонений размеров затворов на ПК от размеров затворов на фоторезистивной маске лучшие результаты были получены на пластине № 4 (уход размеров не превышал 1 мкм), тогда как на пластине № 5 он составил в среднем порядка 1,5 мкм.

## Выводы

1. Обнаружена корреляция между длиной затвора МОП-тразисторов в КМОП ИС и процентом выхода годных микросхем с пластины.
2. Разброс длин затворов МОП-транзисторов по площади пластины находится в пределах 10–12,5 мкм.
3. Разброс длин затворов МОП-транзисторов по площади пластины обусловлен неравномерностью травления ПК, что связано с медленным удалением продуктов травления со средних областей пластин и недостаточным притоком свежего травителя.
4. Введение в технологический процесс травления ПК перемешивание травителя обеспечивает меньший разброс длин затворов по площади кремниевых пластин.

## Литература

1. Скворцов А.М., Халецкий Р.А. Литография в микроэлектронике. Учебное пособие. СПб: СПб ГИТМО (ТУ), 2003. 80 с.
2. Скворцов А.М. Технология микросхем и элементов ЭВА. Л.: ЛИТМО, 1978. 83 с.

# РАЗРАБОТА ЛАБОРАТОРНОЙ ТЕХНОЛОГИИ ПОЛУЧЕНИЯ НАНОКОМПОЗИТНЫХ ПЛЕНОК НА КРЕМНИЕВЫХ ПОДЛОЖКАХ

**И.И. Стройков**
**Научный руководитель – д.т.н., профессор А.М. Скворцов**

В работе исследуется возможность формирования нанокомпозитных пленок на основе кремния, диоксида кремния и различных силикатных стекол. Пленки изготавливались на термически окисленных монокристаллических кремниевых пластинах. Разработана лабораторная технология получения порошков материалов, из которых готовится суспензия, нанесения суспензии на подложку и формирования пленок путем спекания.

## Введение

Проблема получения тонкодисперсных порошков металлов, сплавов, соединений и сверхмелкозернистых материалов из них, предназначенных для различных областей техники, давно обсуждается. В последнее десятилетие интерес к этой теме существенно возрос, так как обнаружилось, что уменьшение размера частиц материала ниже некоторой пороговой величины может проводить к значительному изменению свойств. Такие эффекты появляются, когда средний размер кристаллических зерен не превышает 100 нм, и наиболее отчетливо наблюдается, когда размер зерен менее 10 нм [1].

Научный интерес к нанокомпозитным структурам и материалам связан, прежде всего, с ожиданием различных размерных эффектов на свойствах наночастиц или наноструктур, размеры которых соизмеримы или меньше, чем характерный корреляционный масштаб того или иного физического явления или характерная длина, фигурирующие в теоретическом описании какого-либо свойства или процесса (например, длина свободного пробега электронов, дебройлевская длина волны, размер магнитного домена в ферромагнетиках и пр.).

Управление фундаментальными свойствами твердых тел (полупроводники, металлы, полимеры и т.д.), основанное на синтезировании в их объеме наноразмерных фаз выделений, кристаллитов, дефектных структур или формировании на поверхности пленочных наноструктур, в настоящее время составляет одну из главных проблем ведущих научных центров мира, работающих в направлении нанотехнологий.

Наночастицы и нанослои широко применяются в производстве современных микроэлектронных устройств. Нанокристаллические материалы представляют собой особое состояние конденсированного вещества – макроскопические ансамбли ультрамалых частиц с размерами до нескольких нанометров. Необычные свойства этих материалов обусловлены как особенностями отдельных частиц (кристаллитов), так и их коллективным поведением, зависящим от характера взаимодействия между наночастицами [2]. Длительное время основное внимание было сосредоточено на изучении малых частиц – нанокластеров, изолированных атомов, а также поликристаллических твердых тел. Создание методов получения компактных материалов с необычной тонкозернистой структурой, в которой зерна имеют нанометровые размеры, позволило перейти к изучению структуры и свойств твердого тела в нанокристаллическом состоянии.

## Экспериментальная часть

В настоящее время основными методами получения компактных нанокристаллических материалов являются компактирование изолированных нанокластеров, получение испарением и конденсацией, осаждением из растворов или разложением из прекурсоров; кристаллизация аморфных сплавов; интенсивная пластическая деформация. В настоящей работе сделана попытка получения нанокомпозитных материалов, состоя-

щих из кремния, диоксида кремния и различных силикатных стекол на монокристаллической кремниевой подложке.

Технологию получения нанокомпозитных (НК) пленок на кремниевых подложках стоит условно разделить на несколько основных этапов:

- приготовление порошков используемых материалов;
- приготовление суспензии на основе полученных порошков;
- нанесение суспензии на подложку;
- формирование пленки нанокомпозита путем спекания.

Для приготовления порошка кремния использовались кремниевые не окисленные пластины, предварительно прошедшие несколько этапов очистки:

1. промывка с использованием чистящих средств (порошок, мыло и т.д.);
2. травление в кипящем растворе калиевой щелочи KOH;
3. промывка в дисцилированной воде;
4. сушка в термостате при температуре T = 100 °C.

Получение порошкового кремния производилось с помощью шаровой мельницы. Основа метода шарового размола – механическая обработка твердых смесей, при которой происходят измельчение и пластическая деформация веществ, ускоряется массоперенос, а также осуществляется перемешивание компонентов смеси на атомарном уровне, активируется химическое взаимодействие твердых реагентов. В результате механического взаимодействия в приконтактных областях твердого вещества создается поле напряжений. Релаксация его может происходить путем выделения тепла, образования новой поверхности, возникновения различных дефектов в кристаллах, возбуждения химических реакций в твердой фазе. Схематично этот процесс может быть представлен так, как показано на рис. 1 [3].

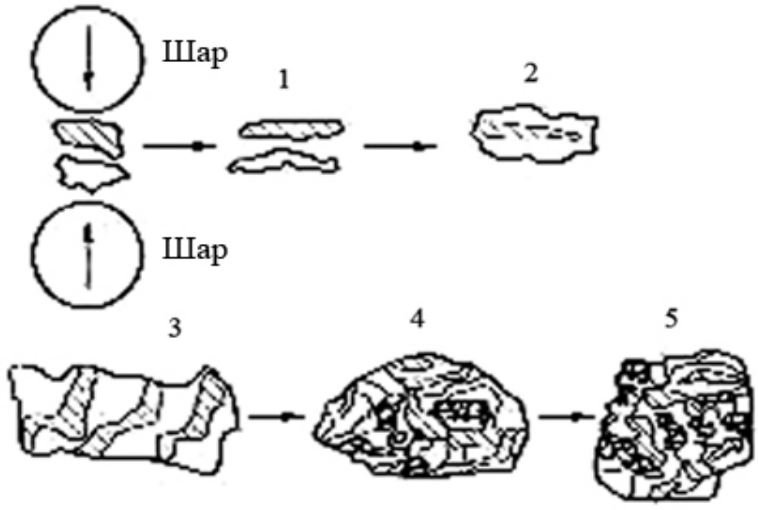



Рис. 1. Схема воздействия на материал во время шарового размола
(1–5 – стадии размола) [3]

Преимущественное направление релаксации зависит от свойств вещества, условий нагружения, размеров и формы частиц. По мере увеличения механического импульса и времени воздействия происходит постепенный переход от релаксации путем выделения тепла к релаксации, связанной с разрушением, диспергированием и пластической деформацией материала и появлением аморфных структур различной природы. Другим каналом релаксации поля напряжения может быть химическая реакция, инициируемая разными механизмами, такими как прямое возбуждение и разрыв связи, реализованные в вершине трещины, локальный тепловой разогрев, безызлучательный распад экситонов и др. [4]

Механический размол – наиболее производительный способ получения больших количеств НК порошков. При механическом размоле порошков деформация первоначально локализуется в полосах сдвига, содержащих большое число дислокаций с высокой плотностью. При достижении определенного уровня напряжений эти дислокации аннигилируют и рекомбинируют с малоугловыми границами, разделяющими отдельные зерна; на этом этапе истирания уже образуются зерна диаметром 20–30 нм, и их количество растет по мере истирания. На следующем этапе истирания ориентация отдельных кристаллов относительно друг друга становится случайной вследствие скольжения по границам зерен.

Простейший аппарат для измельчения – шаровая вращающаяся мельница, представлен на рис. 2. Она представляет собой металлический цилиндрический барабан, внутри которого находятся размольные тела, стальные шары (при изготовлении порошка использовались два больших шара $d$ = 18 мм и три маленьких $d$ = 13 мм) [3].

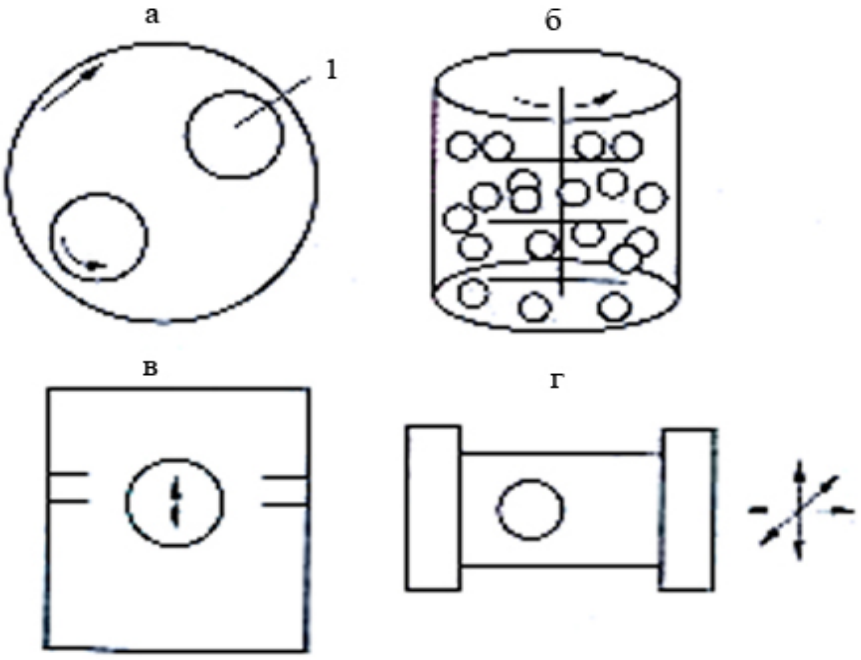



Рис. 2. Типы машин для шарового размола (а – планетарного типа; б – аттриктор; в – одномерная вибрационная машина; г – трехмерная вибрационная машина) [3]

При вращении мельницы размольные тела поднимаются с барабаном (вследствие трения об его стенки) в направлении вращения до тех пор, пока угол подъема не превысит угол естественного откоса, после чего они скатываются или падают вниз и производят измельчение материала, истирая и раздрабливая его между поверхностями мельницы и шаров. При измельчении с помощью шаровой мельницы форма частиц имеет осколочный характер.

Порошок молотился в течение 180 часов.

Следующим этапом технологии является приготовление водяной суспензии.

В коллоидной химии понятие дисперсности включает широкую область размеров частиц. В общем случае высокодисперсные системы называют золями (от лат. Solutio – раствор). Грубодисперсные системы носят название суспензий и эмульсий, в зависимости от характера дисперсной фазы. Суспензии представляют собой микрогетерогенные дисперсные системы с твердой дисперсной фазой и жидкой дисперсионной средой [5].

Суспензии характеризуются кинетической (седиментационной) неустойчивостью. Кинетическая (седиментационная) устойчивость – это способность дисперсной системы сохранять равномерное распределение частиц по всему объему дисперсной фазы. Суспензии являются кинетически неустойчивыми системами. Частицы суспензий по

сравнению с истинными и коллоидными растворами имеют довольно крупные размеры, которые под воздействием силы тяжести обладают способностью к седиментации, т.е. опускаются на дно или всплывают, в зависимости от относительной плотности дисперсной фазы и дисперсионной среды [6].

В качестве основы суспензии могут быть использованы различные щелочи, кислоты, масла и т.д. В нашем случае основой суспензии выбрана дистиллированная вода. Это объясняется тем, что при взаимодействии с водой на поверхности кремния (Si) формируется окисная пленка ($SiO_2$). Таким образом, порошинки кремния будут покрываться и объединяться между собой окислом.

В данной работе в качестве порошковой смеси использовались порошки кремния и фосфатного стекла (полученного таким же способом измельчения, как и кремниевый порошок), в соотношении 1:1. Смешивание порошков производилось в фарфоровой ступке – истиранием с помощью фарфоровой палочки. Затем добавлялась дистиллированная вода, и содержимое перемешивалось. Суспензия осаждалась в течение 5 минут, чтобы наиболее крупные частицы кремния и стекла опустились на дно.

С помощью шприца (пипетки) полученная водяная взвесь наносится на поверхность подложки. В качестве подложки в работе используется окисленная кремниевая пластина с толщиной окисла 1,3 мкм, предварительно прошедшая этап очистки. Далее следует просушка полученного образца для удаления влаги с поверхности, в термостате при температуре T = 100 °C в течение 30–40 минут.

Следующим этапом является спекание. Сырые заготовки имеют рыхлую и недостаточно однородную структуру пленки и структурно обособленные частицы. Спекание можно определить как кинетический процесс освобождения дисперсной системы от избыточной энергии дефектов и энергии поверхности частиц. Это типичный случай релаксационного процесса, само протекание которого обусловлено стремлением системы к равновесному (с меньшей энергией) состоянию. Помимо самого спекания, в материале параллельно протекают процессы рекристаллизации, гетеродиффузии, заключающиеся в образовании и миграции межзеренных границ, формирующих структуру пленки. Они также приближают систему к равновесию.

Процесс спекания условно можно разбить на три стадии: начальную, промежуточную и заключительную. Особенность начальной стадии – образование контактных шеек между частицами. В результате поры сложных конфигураций принимают цилиндрическую форму, что сопровождается резким снижением свободной поверхности заготовки. Промежуточная стадия спекания характеризуется уменьшением сечения этих пор, что сопровождается значительной усадкой пленки. На заключительной стадии происходит полное уплотнение пленки нанокомпозита [7].

Спекание производилось в диффузионных печах СДО-3; при этом варьировалось время спекания в пределах 3–5 часов и температура в пределах 1000–1250 °C. Разогрев и остывание подложек происходили вместе с печью.

## Заключение

В результате проведения работы на кремниевых монокристаллических подложках получены пленки нанокомпозитов разного состава, структуру и свойства которых предполагается в дальнейшем исследовать.

## Литература

1. Гусев А.И., Ремпель А.А. Нанокристаллические материалы. М.: ФИЗМАТЛИТ, 2001. 224 с.
2. http://rvs.itsoft.ru/publications/

3. Валиев Р.З., Грабовецкая Г.П., Колобов Ю.Р. Зернограничная диффузия и свойства наноструктурных материалов. Новосибирск: Наука, 2001. 232 с.
4. Болдырев В.В. Механохимия и механическая активация твердых веществ. // Успехи химии. 2006. 75(3). С. 203–216.
5. Гладков С.О. Физика композитов: Термодинамические и диссипативные свойства. М.: Наука, 1999. 330 с.
6. http://referatw.ru/cgi-bin/main.cgi?level=6&p1=89&p2=63&p3=7771
7. Андреев В.Г. и др. Проблемы порошкового материаловедения. Ч. III. Реология дисперсных систем в технологии функциональной магнитной керамики. Екатеринбург: УрО РАН, 2003. 148 с.

# 3 БИОТЕХНИЧЕСКИЕ ИЗМЕРИТЕЛЬНО-ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ

## ВЫЯВЛЕНИЕ РЕАКЦИЙ ЛЮДЕЙ ПРИ ВОСПРИЯТИИ МУЗЫКИ МЕТОДОМ ГРВ

**А.Ю. Гришенцев, Е.В. Исаева, Е.Н. Петрова, А.В. Шапин**
**Научный руководитель – д.т.н., профессор К.Г. Коротков**

Проведено исследование по выявлению реакций людей при восприятии музыки и синемафонии (исполнение симфонии сопровождается показом кинофильма) в следующих режимах: слушатель (прослушивание музыки), зритель (просмотр фильма без музыки), свидетель (просмотр синемафонии).

### Введение

Свечение объектов различной природы в электромагнитных полях высокой напряженности было обнаружено более 200 лет назад и с тех пор постоянно привлекало внимание исследователей [1]. Однако только с созданием программно-аппаратных комплексов газоразрядной визуализации (ГРВ) в 1995 году исследование этих свечений получило статус научного направления. С тех пор были детально исследованы физические механизмы формирования свечений [2]. Было показано, что характеристики свечения поверхности кожного покрова человека зависят, в первую очередь, от активности вегетативной нервной системы с учетом системы адаптационных уровней [3].

Программно-аппаратные ГРВ биоэлектрографические комплексы нашли практическое применение в следующих основных областях:

- медицина – для анализа состояния вегетативной нервной системы и мониторинга реакций организма в процессе проводимой терапии [4];
- спорт – для оценки уровня соревновательной готовности спортсменов [5];
- исследование жидкостей и материалов – для выявления отличия натуральных и синтетических масел [6], оценки качества косметических препаратов [7], волос человека [8], опасности аллергенов по параметрам ГРВ свечения образцов крови и целого ряда других приложений.

Данная работа посвящена исследованию по практическому применению метода газоразрядной визуализации для выявления реакций людей при восприятии музыкального произведения и синемафонии. Основные задачи:

- выявление критических интервалов реакции и корреляция этих областей у отдельных людей и в целом по группе;
- выявление амплитуды реакции в критических областях;
- сопоставление амплитуд Слушатель + Зритель => Свидетель;
- выявление эмоциональных резонансов и оценка уровня психической энергии для перехода между состояниями Слушатель + Зритель => Свидетель;
- прямой синтез слух + зрение – топографическое смещение внимания.

### Физические принципы формирования изображений при газоразрядной визуализации

Для понимания принципов работы приборов газоразрядной визуализации (ГРВ) рассмотрим принципиальную схему устройства прибора (рис. 1).

Исследуемый объект помещается на поверхности диэлектрической (в большинстве модификаций – кварцевой) пластины, на которую подаются импульсы напряжения от импульсного генератора, для чего на обратную сторону пластины нанесено прозрачное

токопроводящее покрытие. При высокой напряженности поля в газовой среде пространства контакта объекта и пластины развивается разряд в газовой фазе, носящий название «скользящий газовый разряд» [9], параметры которого определяются свойствами объекта. Свечение разряда 6 с помощью оптической системы и ПЗС-камеры преобразуется в видеосигналы, которые поступают в виде одиночных кадров или серии кадров в компьютер. Специализированный программный комплекс позволяет провести обработку изображений (ГРВ-грамм), представляющих собой пространственное распределение освещенности, зависящее от состояния исследуемого объекта.

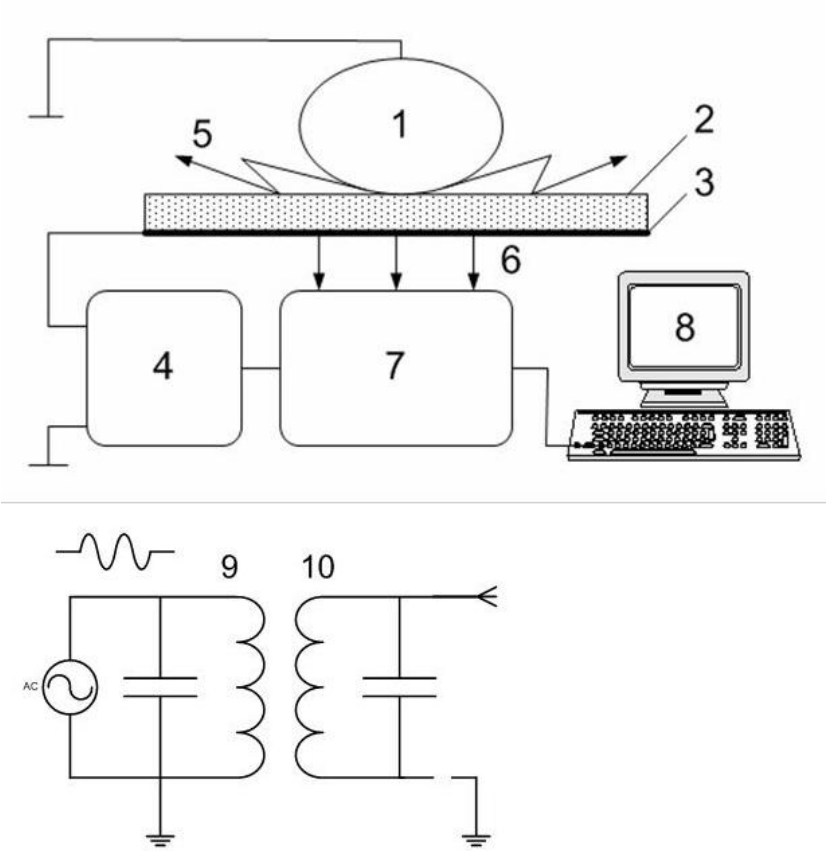
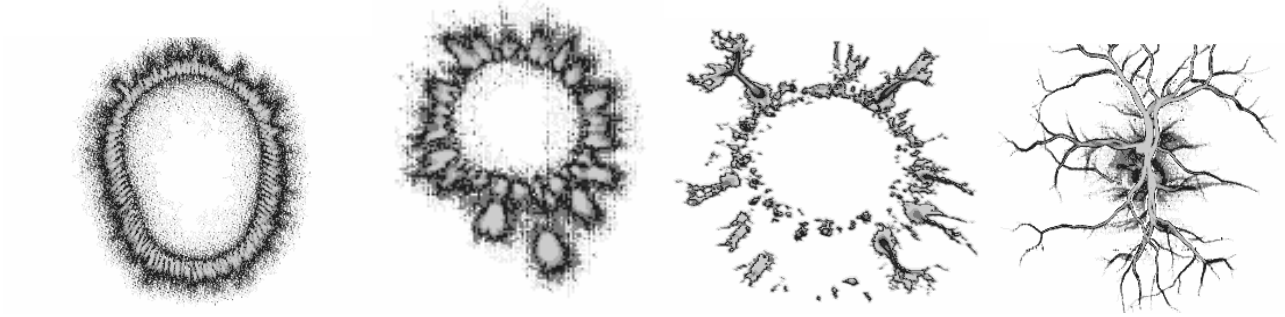



Рис. 1. Принципиальная схема метода газоразрядной визуализации (ГРВ):
1 – исследуемый объект; 2 – диэлектрическая пластина (кварц); 3 – прозрачное токопроводящее покрытие; 4 – генератор импульсов; 5 – скользящий газовый разряд; 6 – свечение разряда; 7 – оптическая система и ПЗС-камера; 8 – компьютер; 9, 10 – система связанных LC контуров, образованных элементами схемы прибора и эквивалентной емкостью исследуемого объекта



а б в г

Рис. 2. Примеры ГРВ-грамм: а – палец руки практически здорового человека;
б – палец руки кардиологического больного; в – палец руки человека
в стрессе; г – капля жидкости

Для примера на рис. 2 приведены изображения газоразрядного свечения пальцев рук человека, типичные для различных состояний здоровья. У практически здорового человека (рис. 2а) свечение равномерное и яркое по всей окружности пальца. У больного (в данном примере кардиологического) свечение характеризуется меньшей яркостью и наличием провалов свечения (рис. 2б), в то время как у человека в состоянии физиологического стресса (рис. 2в) это свечение имеет только ряд отдельных выбросов. На этом же рисунке приведено характерное свечения капли жидкости (рис. 2г), несущее информацию о ее физико-химических свойствах.

Анализ больших баз данных людей в различном состоянии позволил с помощью методов компьютерной обработки изображений определить комплекс параметров, которые в ряде случаев позволяют формировать диагностические заключения. В специализированном программном ГРВ-комплексе вычисляются следующие параметры: площадь изображения (количество пикселей изображения ненулевой яркости), распределение пикселей изображения по яркости и ряд других, описываемых ниже.

При всем многообразии конкретных технических решений сущность процесса визуализации может быть сведена к некоторой теоретической схеме [2]. Первичным процессом является процесс взаимодействия электромагнитного поля (ЭМП) с объектом исследования, в результате которого при определенной напряженности ЭМП с поверхности объекта возникает эмиссия заряженных частиц и фотонов, участвующих в инициировании начальных фаз газового разряда. Газовый разряд, в свою очередь, может влиять на состояние объекта, вызывая вторичные эмиссионные и тепловые процессы. Неоднородность поверхности и объема исследуемого объекта, процессы эмиссии заряженных частиц или выделения газов оказывают влияние на параметры электромагнитного поля, за счет чего изменяются характеристики тока разряда и оптического излучения. При этом основная информация извлекается из характеристик свечения. Приемник излучения преобразует пространственное распределение освещенности в изображение, анализ которого приводит к формированию набора параметров. Из параметров строится симптомокомплекс, необходимый для формирования заключения: анализа состояния пациента при конкретном заболевании, количественной оценки уровня психоэмоциональной реакции испытуемого на воздействующие стимулы, оценки уровня стресса и так далее. Дополнительная информация извлекается из анализа динамических рядов изображений, т.е. временной динамики процессов [10].

## Методика эксперимента

Эксперимент проводился в три последовательных этапа. Была набрана группа из 10 человек. На первом этапе испытуемые прослушивали седьмую симфонию Д.Д. Шостаковича, на втором просматривали специально подобранную документальную кинохронику режиссера Г. Параджанова, а на третьем им была представлена синемафония (классическое исполнение седьмой симфонии Д.Д. Шостаковича с показом документальной кинохроники).

Запись данных проводилась со среднего пальца правой руки испытуемых в течение всего времени исполнения музыки, а также 3 минут до начала и 3 минут после окончания музыкального произведения. Использовались приборы «ГРВ Компакт», работающие в автоматическом режиме: каждые 15 секунд подавался импульс напряжения, и проводилась запись сигнала свечения с пальца исследуемого в компьютер. По окончании каждого эксперимента все данные обрабатывались: вычислялись параметры свечения и строились временные ряды этих параметров.

Для определения статистически значимых точек реакции исследуемых для каждого участка симфонии вычислялись средние значения сигнала по временному участку, и определялись 25 % и 75 % перцентали. Вычислялись все точки, имеющие значения, выходящие за величины перцентилей. Эти точки представляли собой статистически

значимые отклонения от среднего уровня с учетом вариаций, т.е. статистически значимые реакции психофизиологического состояния исследуемых на музыкальные моменты. Значения реакции могли быть как положительны, так и отрицательны, в обоих случаях они характеризовали эмоционально значимые моменты для данного человека. Строились графики временных зависимостей этих отклонений, вычислялись линии тренда путем усреднения по каждым 5 точкам. Такая обработка позволяет наглядно представить кривую изменения энергетики человека и оценить выраженность психоэмоциональной реакции в различных частях симфонии.

### Алгоритм построения графиков

Построение индивидуального графика эмоциональная реакция каждого исследуемого осуществлялось по следующей схеме.

- Полученная серия ГРВ-грамм обрабатывалась в программе «GDV Scientific Laboratory» с формированием файла числовых параметров.
- Полученные данные (сформированный файл) переносились в программу Excel.
- Из вычисленных параметров для дальнейшего анализа использовалась только площадь изображения.
- Вычислялась площадь свечения пальца в относительных величинах по формуле

$$\mathrm{A}_{отн} = \mathrm{A} \big/ \mathrm{A}_{ср}, \qquad (1)$$

где $\mathrm{A}_{отн}$ – площадь свечения в относительных величинах, $\mathrm{A}$ – площадь свечения в пикселях, $\mathrm{A}_{ср}$ – средняя площадь свечения по рассматриваемой части симфонии.

- Строился график $\mathrm{A}_{отн}(t)$, где $t$ – время по хронометражу.
- На построенном графике строилась линия тренда по формуле

$$Ft = \frac{A_t + A_{t-1} + \ldots + A_{t-n+1}}{n}, \qquad (2)$$

где $Ft$ – значение точки линии тренда, $\mathrm{A}_t$ – значение точки ряда данных, $n$ – значение периода.

Построение усредненного графика эмоциональной реакции по каждой группе (слушатель, зритель, свидетель) проводилось в следующем порядке.

- Для испытуемого находились точки, выходящие за 25 % и 75 % перцентили.
- Вычислялось отклонение этих точек от перцентилей и сводилось в отдельную таблицу для всей группы. Отклонение от перцентилей находилось по следующему алгоритму:

```
If area > per 75 Then
G = area – per 75
End If
If area < per 25 And area <> 0 Then
G = area – per 25
End If,
```

где *area* – значение площади свечения в относительных единицах, *per* 25 и *per* 75 – значение 25 % и 75 % перцентили, $G$ – значение отклонения от перцентили.

- В таблице находилось среднее значение отклонений в каждый момент времени по группе (по каждой строчке).
- Строился график $F_{ср}(t)$, где $t$ – время по хронометражу.

Построение сравнительного графика количества реакций в группе в процентах (реакцией в данный момент времени назвали превышение сигнала выше 75 % или ниже 25 % перцентили) проводилось по следующей схеме.

- По группе находили процент реакций в каждый момент времени.
- Для сопоставления данных для разных режимов измерения на одной координатной оси строили графики зависимости процента реакций каждой группы от времени.

**Результаты исследования**

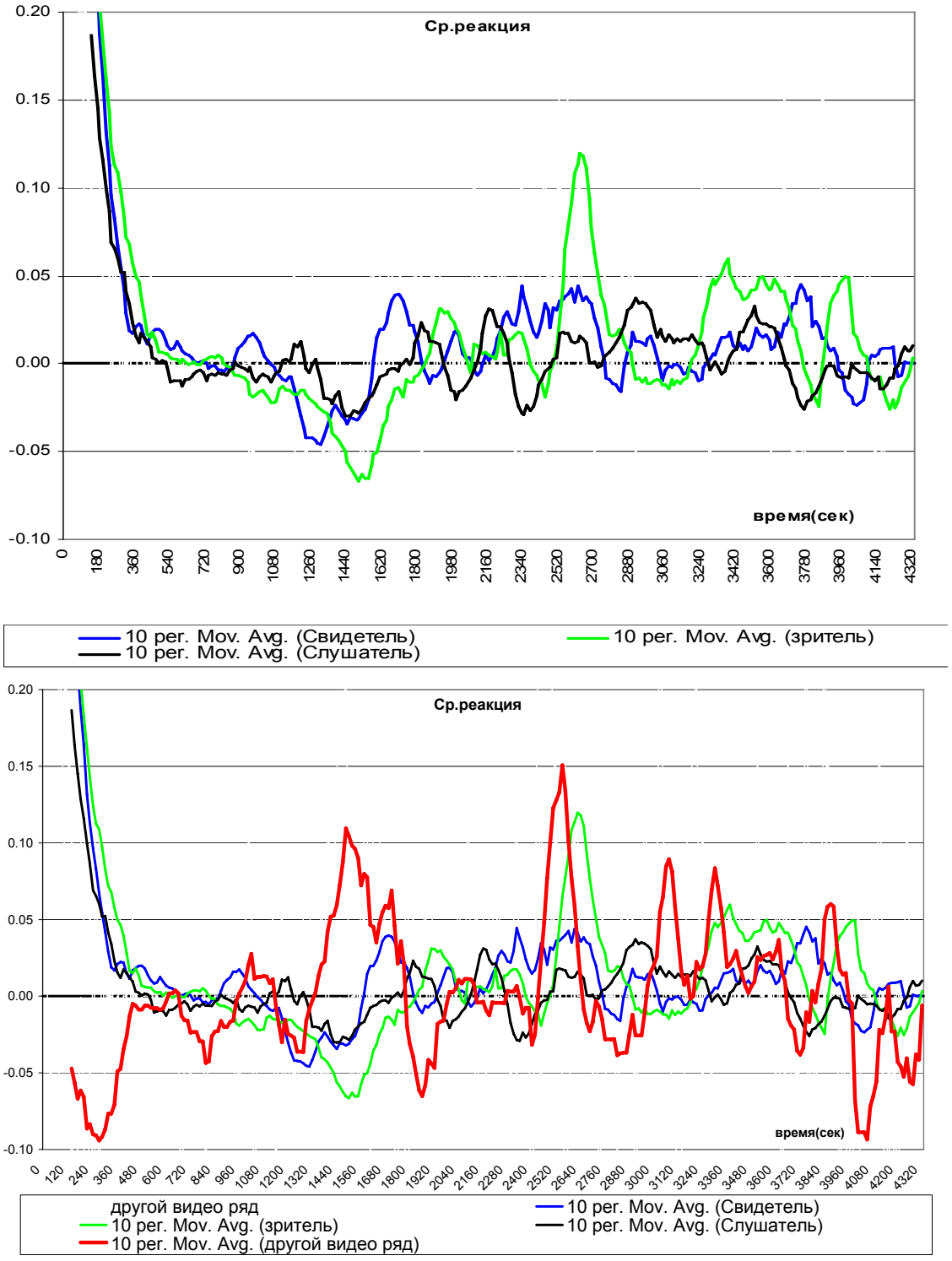



Рис. 3. Временная зависимость отклонений от перцентилей площади свечения во время прослушивания симфонии с графиками усреднения по 5 точкам в различных режимах

Одним из направлений исследования было выявление специфичности созданного видеоряда. Для этого был проведен эксперимент по регистрации реакции той же группы испытуемых на музыку седьмой симфонии Шостаковича при одновременном просмотре другого видеоряда. На рис. 3 приведены графики реакций по всему времени симфонии для трех режимов и с наложением графика реакций при другом видеоряде. Анализ этих графиков позволяет сделать следующие выводы.

1. В начальной фазе, примерно до 1600 с, реакция на другой видеоряд существенно отличается от реакции на синемафонию.
2. В области 1700–2000 с наблюдаются согласованные пики реакции, в основном с режимом «свидетель».
3. Наиболее выраженная реакция, хорошо согласованная как со Зрителями, так и со Свидетелями, наблюдается в области 2400–3000 с. Этой области соответствует сильная эмоциональная реакция.
4. В последних частях симфонии можно отметить некоторое соответствие реакций, однако происходящее со значительным сдвигом во времени.
5. Практически не наблюдается корреляций с реакциями Слушателей.

Эти результаты могут быть интерпретированы как доказательства сильного влияния эмоционально значимых моментов музыки Шостаковича, даже при наличии раздражающего влияния «постороннего».

## Заключение

В ходе проведенных исследований были зафиксированы реакции большой группы испытуемых на воздействие музыки седьмой симфонии Шостаковича в режимах прослушивания музыки (Слушатель), просмотра видеоматериала без музыки (Зритель) и синемафонии (Свидетель). На первом этапе был исследован ряд испытуемых при произвольном характере чередования режимов. Анализ данных показал, что для адекватных выводов необходимо провести обследование группы испытуемых при смене режимов в определенном порядке. Анализ экспериментальных данных для обеих групп позволяет сделать следующие заключения:

1. Разработанный метод регистрации и обработки экспериментальных данных позволяет выявлять статистически значимые критические интервалы реакции испытуемых на воздействие музыки и/или видеоряда.
2. Количество критических интервалов зависит от выбора уровня значимости реакции путем определения граничной перцентили. Экспертным путем выбраны 25 % и 75 % перцентили. Это означает, что значимыми признаются реакции, превышающие на 75 % средний уровень вариабельности сигнала в обе стороны.
3. Критические интервалы ранжируются по степени реакции в группе (процент среагировавших в данном интервале).
4. Выявлены амплитуды реакции в критических интервалах, как индивидуальные, так и усредненные по группе.
5. Показано, что статистически значимые критические интервалы реакции коррелируют у различных испытуемых между собой и по отношению к групповым реакциям.
6. Сопоставление реакций в режимах Слушатель => Зритель => Свидетель показало наличие большого количества согласованных критических интервалов реакции, в ряде случаев сдвинутых относительно друг друга по времени.
7. Этот факт свидетельствует об объективном характере воздействия музыки седьмой симфонии Шостаковича вне зависимости от момента прослушивания и личности испытуемого.

8. Эмоциональные реакции Свидетелей, как правило, были более сильными по амплитуде по сравнению со Слушателями. Это свидетельствует о комплексном характере воздействия синемафонии через различные сенсорные системы с формированием эмоциональных резонансов на физиологически значимом уровне.

9. У Зрителей наблюдался ряд согласованных резонансов по сравнению со Свидетелями. Это может быть интерпретировано как свидетельство высокой степени корреляции эмоционального содержания видеоряда и музыкальных фраз.

10. Амплитуда эмоциональных резонансов Зрителей в ряде случаев превышала амплитуду реакций в других режимах. Это может быть интерпретировано как признак концентрации внимания на одной сенсорной системе (видеоряд) при отсутствии других сенсорных раздражителей.

11. Можно говорить о прямом синтезе Слух + Зрение, вызывающем топографическое смещение во внимание.

12. Введение случайного видеоряда приводило к появлению целого ряда эмоциональных реакций, не совпадающих с реакциями в исследованных режимах. Это подтверждает вывод об усилении эмоциональных резонансов в режиме синемафонии.

13. Разработанный подход может быть использован для исследования влияния музыкальных произведений на психоэмоциональное состояние человека.

14. Полученные данные хорошо согласуются с опубликованными результатами экспериментов по приборной регистрации психоэмоциональных реакций человека на музыку.

## Литература


1. Коротков К.Г. Эффект Кирлиан. СПб: Ольга, 1995. 218 с.
2. Коротков К.Г. Основы ГРВ биоэлектрографии. СПб: СПбГУ ИТМО, 2001. 356 с.
3. Полушин Ю.С., Струков Е.Ю., Широков Д.М., Коротков К.Г. Возможности метода газоразрядной визуализации в оценке операционного стресса у больных с абдоминальной хирургической патологией // Вестник хирургии. 2002. Т.161. №5. С. 118.
4. Александрова Р.А., Шульга А.Ф., Петровский И.Д., Галкина О.В., Нутфуллина Г.М., Зайцев С.В., Магидов М.Ю., Пягай Е.И. Результаты лечения больных с мультиморбидной патологией с помощью малых воздействий // Ученые записки СПб ГМУ им. акад. И.П. Павлова. 2002. Т.IX. № 4. С. 75–78.
5. Бундзен П.В., Коротков К.Г., Короткова А.К., Макаренко А.И. Результаты и перспективы использования технологии квантовой биофизики в подготовке высококвалифицированных спортсменов. // Теория и практика физической культуры. 2003. №3. С. 26–43.
6. Бундзен П.В., Коротков К.Г., Короткова А.К., Мухин В.А., Прияткин Н.С. Психофизиологические корреляты успешности соревновательной деятельности спортсменов Олимпийского резерва. // Физиология человека. 2005. Т. 31. № 3. С. 1–9.
7. Коротков К.Г., Крыжановский Э.В., Филатов С.И., Филиппосьянц Ю.Р. Метод выявления лиц, склонных к совершению противоправных действий. М.: ГУ НПО «Специальная техника и связь» МВД России, 2005. С. 32.
8. Дашук П.Н. Скользящий разряд в устройствах газоразрядной визуализации. / Тезисы докладов международного научного конгресса «Наука, Информация, Сознание», СПб, 1999. С. 70.
9. Бабицкий М.А. Автоматизированное проектирование систем анализа динамических газоразрядных изображений. / Автореферат диссертации на соискание ученой степени кандидата технических наук. СПбГУ ИТМО, 2003.

# ИССЛЕДОВАНИЕ ЭЛЕМЕНТНОЙ БАЗЫ НА ОСНОВЕ ТОКОПРОВОДЯЩИХ ПОЛИМЕРОВ ДЛЯ БЛОКА УПРАВЛЕНИЯ МАНИПУЛЯТОРАМИ МЕТОДОМ ГРВ

**В.А. Нечаев**

**Научный руководитель – д.т.н., профессор В.Л. Ткалич**

Приводятся данные об исследовании ГРВ-свечения элементной базы на основе токопроводящих полимеров для блока управления манипуляторами. Разработана методика определения качества используемого материала.

## Введение

Использование высококачественного полимера для изготовления элементной базы на основе токопроводящих полимеров для блока управления манипуляторами является важнейшим этапом их производства. Первостепенное значение в этом случае имеет своевременная оценка физико-химических свойств используемого материала, позволяющая избежать траты времени и средств на использование заведомо непригодного сырья.

Одна из наиболее действенных и, что принципиально, оперативных технологий сверхраннего выявления и диагностики состояния токопроводящих материалов – технология газоразрядной визуализации (ГРВ), осуществляемая с использованием программно-аппаратурного ГРВ-комплекса [1, 2]. Современная аппаратура, применяемая при исследовании ГРВ-методики, дает возможность регистрировать и анализировать газоразрядное свечение, индуцированное у образцов полимеров.

## Схема эксперимента

В ходе исследования было проведено 28 экспериментов с образцами полимеров. Использовались образцы диаметром 4 мм и длиной 4 см. Каждый образец помещался в трубку из политетрафторэтилена. Затем трубка закреплялась в устройство для исследования материалов и устанавливалась на ГРВ-камеру для регистрации серий динамических ГРВ-грамм. Схема эксперимента показана на рис.1.

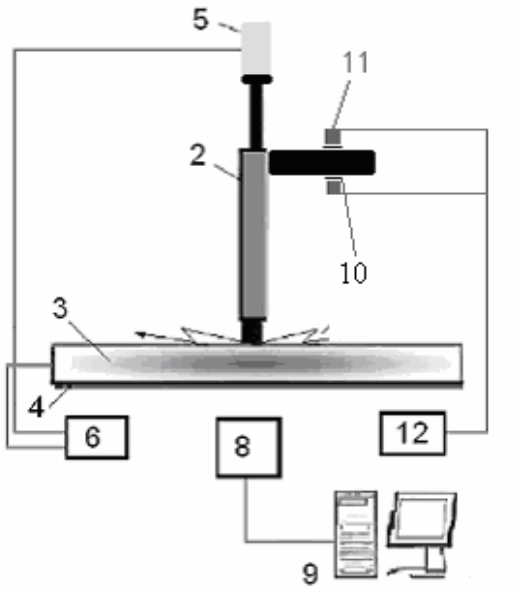



Рис. 1. Принципиальная схема экспериментальной установки: 1 – исследуемый образец полимера; 2 – политетрафторэтиленовая трубка; 3 – диэлектрическая пластина; 4 – прозрачное токопроводящее покрытие; 5 – заземленный металлический стержень; 6 – высоковольтный импульсный генератор; 7 – скользящий газовый разряд; 8 – оптическая система с ПЗС–камерой; 9 – компьютер; 10 – плоско-параллельные электроды; 11 – магнитные катушки; 12 – источник питания 0-200В

Постоянное напряжение от стабилизированного источника 12, подаваемое на электроды 10, последовательно устанавливают на определенных значениях в диапазоне от 0 до 200 В. При каждом значении напряжения образцы выдерживаются в электрическом поле по 10 секунд до начала съемки, а затем снимается 5 avi-файлов, продолжительностью по 5 с. Первичная обработка изображений проводится в программе «GDV SciLab», результатами которой являются таблицы численных значений таких параметров, как площадь свечения и средняя интенсивность свечения. Дальнейшая обработка и построение графиков проводится в программе «Microsoft Excel».

## Результаты эксперимента

В программе «Microsoft Excel» были построены зависимости площади и интенсивности свечения образца от электрического поля, примеры которых представлены на рис. 2, 3.

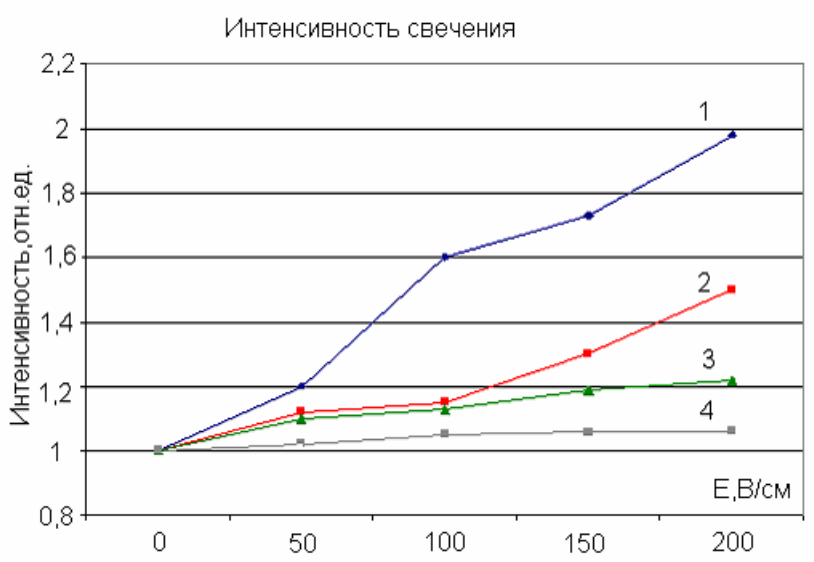



Рис. 2. Зависимости интенсивностей свечения образца от напряженности электрического поля в зависимости от степени полимеризации: 1 – *n*>10000, 2 – *n*=10000, 3 – *n*=4000, 4 – *n*>2000

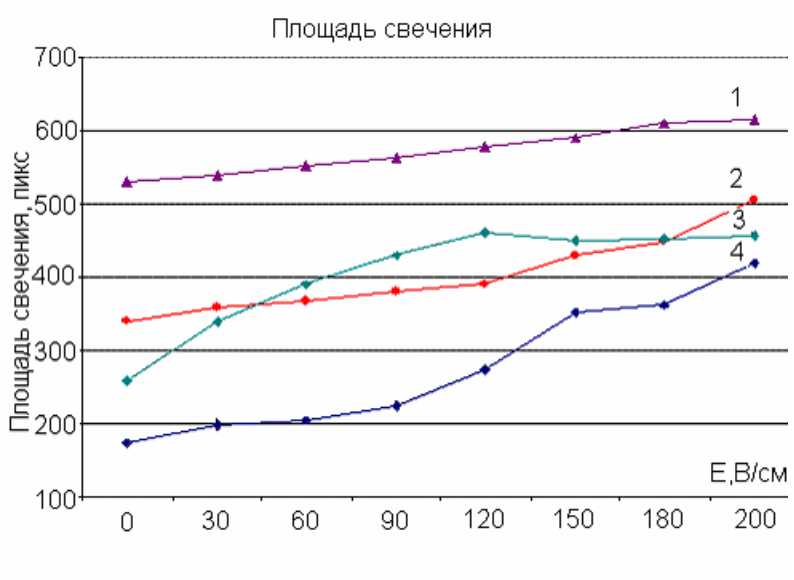



Рис. 3. Зависимости площадей свечения различных образцов от напряженности поля: 1, 2, 3, 4 – порядковый номер образца

Как видно из рис. 2, с уменьшением степени полимеризации зависимость интенсивности свечения полимера от прикладываемого электрического поля уменьшается, а при значениях *n* менее 2000 зависимость от поля пропадает. Поэтому в дальнейших исследованиях будут браться полимеры со степенью полимеризации $n>10000$. Как видно из рис. 3, зависимость площади свечения полимера от напряженности поля сильно отличается для различных образцов. В большинстве случаев значения амплитуд интенсивности и площади свечения образцов увеличивались при увеличении электрического поля, однако в ряде случаев эти зависимости имели спадающий характер.

## Заключение

На основании полученных результатов можно утверждать, что электрическое поле является тестирующим фактором при анализе ГРВ-свечения полимеров. Есть основания предполагать, что реакция площади и интенсивности свечения полимеров на повышение электрического поля зависит от их физико-химического состояния. Этот вопрос требует дальнейшего исследования.

## Литература


1. Korotkov K.G. Measuring energy filds: state of the Science: GDV Bioelectrografhy series Vol. I, Ed., Backbone Publishing Co. Fair Lawn, USA, 2004.
2. K. Korotkov, D. Korotkin. // J. Appl. Physics. 2001. V. 89. P. 4732.
3. Нечаев В.А., Петрова Е.Н. Исследование ГРВ – свечения волос под воздействием электрического поля. / Наука. Информация. Сознание: тезисы IX международного конгресса по ГРВ биоэлектрографии. СПб, 2005.
4. Коротков К.Г., Нечаев В.А., Петрова Е.Н., Вайншелбойм А, Коренюгин Д.Г., Шигалев В.К. Исследование ГРВ свечения волос. // Приборостроение. 2006. Т.49. № 2. С. 51–57.

# ФИЗИКО-МЕХАНИЧЕСКИЕ СВОЙСТВА ЭЛЕМЕНТНОЙ БАЗЫ НА ОСНОВЕ ТОКОПРОВОДЯЩИХ ПОЛИМЕРОВ ДЛЯ БЛОКА УПРАВЛЕНИЯ МАНИПУЛЯТОРАМИ

**В.А. Нечаев**

**Научный руководитель – д.т.н., профессор В.Л. Ткалич**

В статье приводятся метод определения физико-механических свойств элементной базы на основе токопроводящих полимеров для блока управления манипуляторами. Проведены исследования пленок с улучшенными физико-механическими свойствами.

## Введение

Актуальной задачей изготовления элементной базы для блока управления манипуляторами является использование современных синтетических материалов, обладающих высокой прочностью и хорошей износостойкостью. Большое значение имеет подбор материала для изготовления сенсорных элементов, так как они подвергаются постоянным воздействием оператора, а блок управления должен обладать безотказностью работы и достаточно высокой надежностью при постоянном его использовании.

Определить надежность полимера, используемого в сенсорах, можно по физико-механическим свойствам [1, 2], наиболее точно ее можно выявить с помощью диаграммы прочности при разрыве и по показателю текучести расплава.

## Схема эксперимента

Исследования проводились на разрывной машине Zwick 1435. Образцы изготавливались по ГОСТ 14236-81: из различных мест пленки толщиной 40–50 мкм с помощью канцелярского ножа и линейки вырезались 5 полосок вдоль и 5 поперек рукава размерами 15×100 мм. Всего использовались 11 видов пленок с последующим увеличением на 0,2 % специальной добавки, которая должна изменять ее физико-механические характеристики. С помощью скотча образцы устанавливались в зажимы, затем их растягивала траверса, после разрыва полимера траверса возвращалась в первоначальное положение, принтер, присоединенный к разрывной машине, печатал значения относительного удлинения и прочности при разрыве.

Для измерения показателя текучести расплава использовался ИИРТ – 4 ПТР, опыты проводили по ГОСТ 11645-73 , при условиях 2,16 кг /190 °С и 5 кг /190 °С. Пленка сворачивалась в плотный рулон, заталкивалась в материальный цилиндр ИИРТа, после термостатирования шток с грузом отпускался из зажима, стренги выдавленного за определенный промежуток времени материала срезались и взвешивались на электронных лабораторных весах, далее рассчитывалась масса материала, вытекшего за 10 минут, полученные значения усреднялись.

## Результаты эксперимента

Обработка данных с расчетом погрешности, построении линии тренда и уравнения кривых, а также достоверность аппроксимации была проведена в пакете Microsoft Excel. На рис. 1–4 представлены зависимости, полученные по усредненным значениям из пяти испытаний.

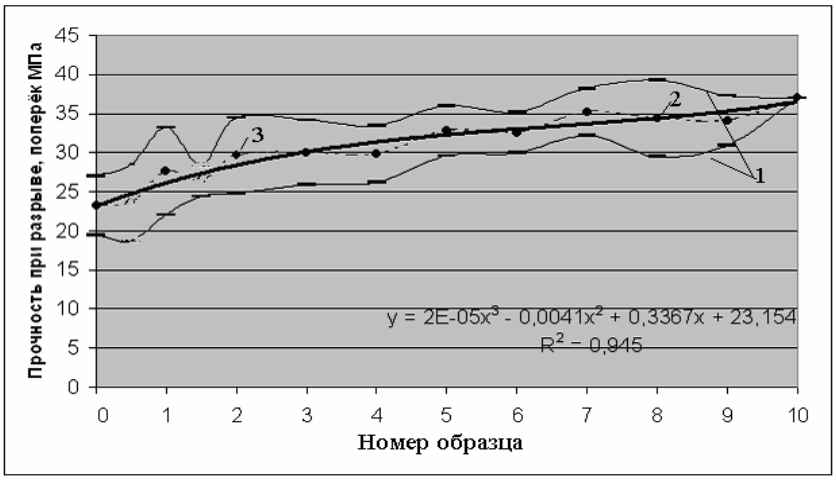



Рис. 1. Прочность при разрыве поперек: 1 – доверительные интервалы, 2 – линия тренда, 3 – кривая по полученным данным

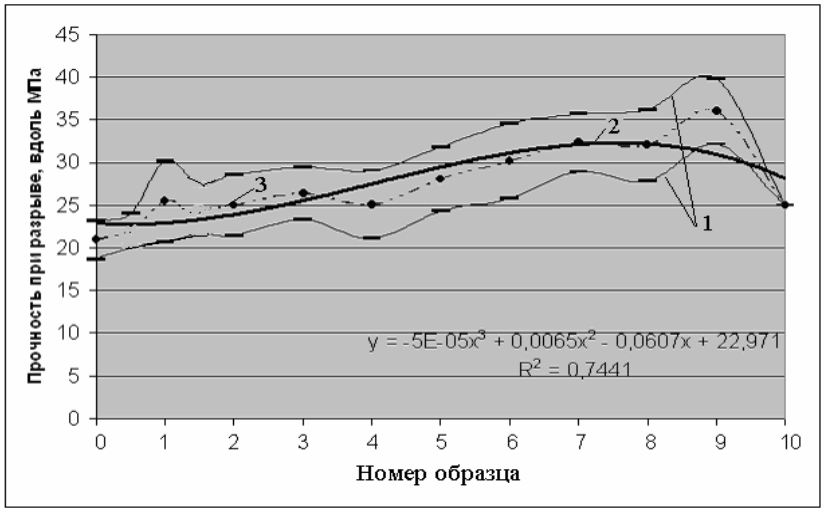



Рис. 2. Прочность при разрыве вдоль: 1 – доверительные интервалы, 2 – линия тренда, 3 – кривая по полученным данным

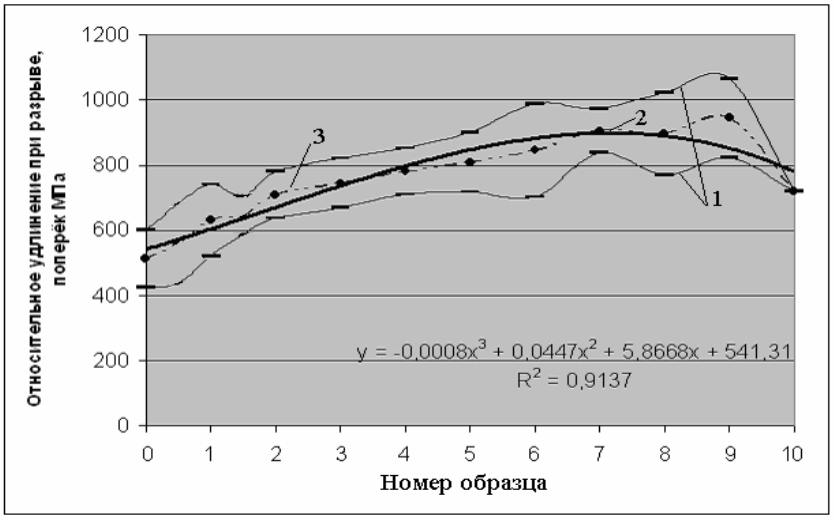



Рис. 3. Относительное удлинение при разрыве поперек: 1 – доверительные интервалы, 2 – линия тренда, 3 – кривая по полученным данным

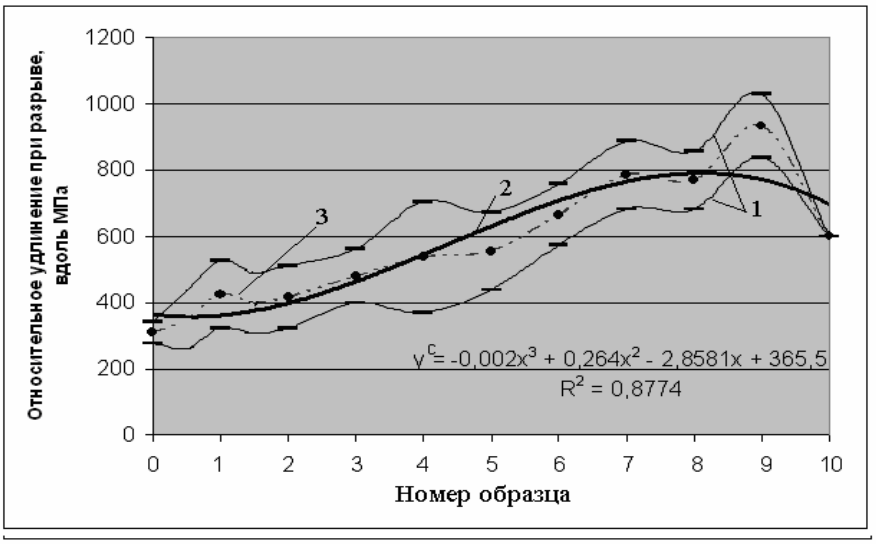



Рис. 4. Относительное удлинение при разрыве вдоль: 1 – доверительные интервалы, 2 – линия тренда, 3 – кривая по полученным данным

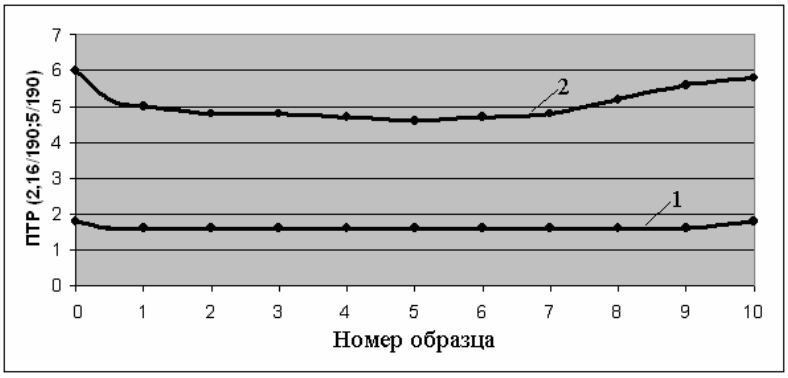



Рис. 5. Показатели текучести расплава: 1 – значения ПТР в условиях 2,16/190; 2 – значения ПТР в условиях 5/190

Как видно из графиков, при повышении содержании добавки в пленках происходит повышение физико-механических свойств, ПТР в условиях 2,16/190 практически никак не изменяется (рис. 5), в условиях 5/190 происходит небольшое падение ПТР, а при содержании добавки свыше 1,6 % происходит плавное его увеличение до прежнего значения.

## Заключение

Эксперименты показали, что при введении 1,8 % специальной добавки полимер обладает лучшими физико-механическими свойствами. Следовательно, сенсорная пленка, изготовленная из него, будет обладать большей надежностью за счет более высокого относительного удлинения при относительно высоких значениях прочности при разрыве. Кроме того, можно заключить, что полимер, используемый для изготовления элементной базы, обладает гомогенными свойствами в продольном и поперечном сечении, что говорит о его хорошей износостойкости.

## Литература

1. Abdel-Bary E.M. Handbook of Plastic Films. / Rapra Technology limited. Shawbury, Shrewsbury, Shropshire, SY4 4NR, United Kingdom, 2005. P. 351.
2. Бристон Дж.Х, Катан Л.Л. Полимерные пленки. Пер. с англ. М.: Химия, 1993. 381 с.

# ОРГАНИЗАЦИЯ ОБМЕНА ДАННЫМИ ПО ШИНЕ USB В ОПЕРАЦИОННОЙ СИСТЕМЕ WINDOWS XP С ПРИМЕНЕНИЕМ ЭЛЕКТРОННЫХ КОМПОНЕНТОВ ФИРМЫ FTDI

**А.Ю. Гришенцев, Е.Н. Петрова**
**Научный руководитель – д.т.н., профессор К.Г. Коротков**

Данная статья посвящена организации обмена данными между персональным компьютером и внешним устройством по шине USB на базе электронных компонентов фирмы FTDI. Рассмотрена схема установки драйверов и концепция реализации программной поддержки.

## Введение

Применение аппаратно–программных комплексов в современных условиях подразумевает использование новых стандартных протоколов обмена между ПК (персональным компьютером) и внешним ЭУ (электронным устройством). Такими протоколами на сегодняшний день является FireWire, называемый также 1394, и USB. В данной статье рассмотрен вопрос программной реализации обмена данными между ПК и ЭУ с применением электронных компонентов фирмы FTDI (Future Technology Devices International Ltd.), поддерживающими стандарт USB2.0.

## Выбор протокола обмена

Последовательные шины позволяют объединять множество устройств, используя всего 1–2 пары проводов [1]. Функциональные возможности этих шин гораздо шире, чем у традиционных интерфейсов локальных сетей, – USB и FireWire способны передавать изохронный трафик аудио- и видеоданных. Последовательные шины по своей организации сильно отличаются от параллельных. В последовательных шинах нет отдельных линий для данных, адреса и управления – все протокольные функции приходится выполнять, пользуясь одной или двумя (в FireWire) парами сигнальных проводов. Это накладывает отпечаток на построение шинного протокола, который в последовательных шинах строится на основе пересылок *пакетов* – определенным образом организованных цепочек бит.

Наибольшую популярность имеют шины USB и FireWire, хотя последняя пока что в PC-совместимых компьютерах используется не повсеместно. Последовательные шины FireWire и USB, имея общие черты, являются, тем не менее, различными технологиями. Обе шины обеспечивают простое подключение большого числа ЭУ (127 для USB и 63 для FireWire), допуская коммутации и включение/выключение устройств при работающей системе. По структуре топология обеих шин достаточно близка, но FireWire допускает большую свободу и пространственную протяженность. Хабы USB входят в состав многих устройств, и для пользователя их присутствие зачастую незаметно. Обе шины имеют линии питания устройств, но допустимая мощность для FireWire значительно выше. Обе шины поддерживают технологию PnP (автоматическое конфигурирование при включении/выключении) и снимают проблему дефицита адресов, каналов DMA и прерываний.

Шина USB ориентирована на периферийные устройства, подключаемые к PC. Изохронные передачи USB позволяют передавать цифровые аудиосигналы, а шина USB 2.0 способна нести и видеоданные. Все передачи управляются централизованно, и PC является необходимым управляющим узлом, находящимся в корне древовидной структуры шины. Адаптер USB входит в состав всех современных чипсетов системных плат.

Шина FireWire ориентирована на устройства бытовой электроники, которые с ее помощью могут быть объединены в единую домашнюю сеть. К этой сети может быть

подключен компьютер, и даже не один. Принципиальным преимуществом шины 1394 является отсутствие необходимости в специальном контроллере шины (компьютере). Любое передающее устройство может получить полосу изохронного трафика и начинать передачу по сигналу автономного или дистанционного управления – приемники «услышат» эту информацию. При наличии контроллера соответствующее ПО может управлять работой устройств, реализуя, например, цифровую студию нелинейного видеомонтажа или снабжая требуемыми мультимедийными данными всех заинтересованных потребителей информации.

Для связи ПК и ЭУ эффективнее использовать шину USB, так как практически все ПК, поступающие в продажу, оборудованы шиной USB, в то время как шина FireWire на территории России является редкостью в стандартной комплектации ПК.

## Шина USB

Шина USB (Universal Serial Bus – универсальная последовательная шина) появилась по компьютерным меркам довольно давно – версия первого утвержденного варианта стандарта USB1.0 датируется 15 января 1996 года. Разработка стандарта была инициирована весьма авторитетными фирмами – Intel, DEC, IBM, NEC, Northen Telecom и Compaq.

Основная цель стандарта, поставленная перед его разработчиками – создать реальную возможность пользователям работать в режиме Plug&Play с периферийными устройствами. Это означает, что должно быть предусмотрено подключение устройства к работающему компьютеру, автоматическое распознавание его немедленно после подключения и последующей установки соответствующих драйверов. Кроме этого, питание маломощных устройств желательно подавать с самой шины. Скорость шины должна быть достаточной для подавляющего большинства периферийных устройств. Попутно решается историческая проблема нехватки ресурсов на внутренних шинах IBM PC-совместимого компьютера – контроллер USB занимает только одно прерывание независимо от количества подключенных к шине устройств.

Технические характеристики:

- низкая скорость LS (Low Speed USB1.0) – 1,5 Мбит/с;
- полная скорость FS (Full speed USB1.1) – 12 Мбит/с;
- высокая скорость HS (High Speed USB2.0) – 480 Мбит/с;
- максимальное количество подключенных устройств (включая хабы) – 127;
- напряжение питания для периферийных устройств – 5 В;
- максимальный ток потребления на одно устройство – 100 мА;
- допустимый ток потребления от ПК по шине USB – 500 мА;
- шина с использованием промежуточных хабов позволяет соединять устройства, удаленные от компьютера на расстояние до 25 м.

Кабели и разъемы:

Сигналы USB передаются по 4-х проводному кабелю. Сечение концевых разъемов кабеля приведены на рис. 1. Табл. 1 содержит пояснения к назначению использования контактов разъемов и проводов кабеля. Здесь GND – цепь «корпуса» для питания периферийных устройств, VBus – +5V также для цепей питания. Шина D+ предназначена для передачи данных по шине, а шина D – для приема данных.

Кабель для поддержки полной скорости шины (full-speed) выполняется как витая пара, защищается экраном и может также использоваться для работы в режиме минимальной скорости (low-speed). Кабель для работы только на минимальной скорости (например, для подключения мыши) может быть любым и неэкранированным.

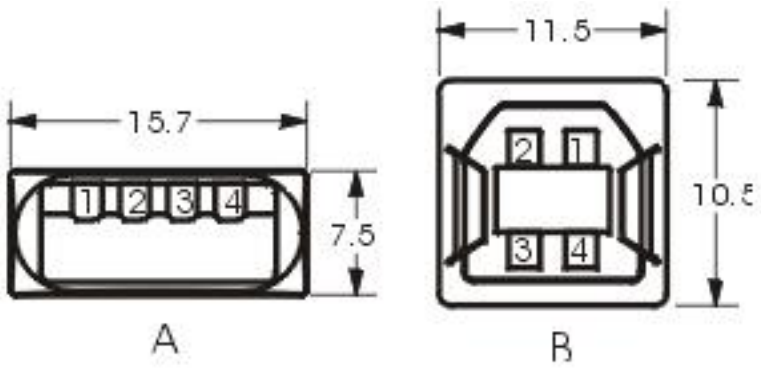



Рис. 1. Сечения разъемов USB кабеля. A – предназначены только для подключения к источнику, т.е. к компьютеру или хабу; B – предназначены только для подключения к периферийному устройству

| Номер контакта | Назначение | Цвет провода |
|---|---|---|
| 1 | V BUS | Красный |
| 2 | D– | Белый |
| 3 | D+ | Зеленый |
| 4 | GND | Черный |
| Оплетка | Экран | Оплетка |

Таблица 1. Назначение контактов и проводов USB кабеля

### Выбор микросхемы

Современный рынок микросхем, поддерживающих обмен данными по шине USB, достаточно обширен. На территории России доступны продукты производства фирм Atmel, FTDI, Cypress, Intel, National Semiconductor и др. Выбор в пользу микросхемы FT232R фирмы FTDI был сделан по следующим причинам:

- возможность перепрограммирования номеров PID и VID, что позволяет избежать конфликтов при использовании микросхемы в различных устройствах;
- полная поддержка протокола обмена по стандарту USB2.0;
- библиотека функций для программной реализации обмена данными со стороны ПК удобна и стабильна в своей работе;
- наличие готовых легко настраиваемых драйверов;
- полная информационная поддержка на сайте фирмы FTDI [4].

### Установка драйверов FT232R

В операционной системе WindowsXP SP1 встроен драйвер VCP (Virtual COM Port) для FT232R, устанавливаемый по умолчанию автоматически, для значений VID 0403 и PID 6001, где VID-Vendor ID, PID-Product ID – код производителя и микросхемы соответственно. В случае автоматической установки драйвера VCP с ЭУ можно работать как с COM-портом, и в этом случае все преимущества скоростного обмена данными по протоколу USB2.0 не используются. Кроме того, наблюдается тенденция сокращения использования COM-портов в ПК, в большинстве современных ноутбуков COM-порт отсутствует. Такая ситуация на рынке говорит о возможном исключении в ближайшем будущем программной поддержки COM-портов. Поэтому целесообразно устанавливать драйвер Ftd2xx USB2.0 устройства, предоставляемый фирмой FTDI [4]. Для этого необходимо удалить драйвер VCP, используя специальную утилиту FTDIUNIN.EXE, ко-

торая входит в состав драйвера Ftd2xx. Делается это следующим образом: файл FTDIUNIN.EXE копируется в папку с драйвером VCP (обычно это C:\WINDOWS\system32) и запускается; далее необходимо выключить и включить кабель USB устройства для повторной инициализации, и в появившемся окне установки драйвера указать папку, в которой находиться драйвер Ftd2xx. В WindowsXP SP2 данная проблема устранена, и при первом включении устройства пользователь может сразу выбрать нужный драйвер.

При установке Ftd2xx выдается сообщение о том, что драйвер не имеет сертификата, это сообщение вызвано тем, что фирма Microsoft не включила в реестр WindowsXP драйвер Ftd2xx. Данное сообщение можно игнорировать, выбрав кнопку продолжения инсталляции.

## Программирование FT232R

Для предотвращения конфликтных ситуаций между устройствами, использующими микросхемы FTDI, подключенными к одному ПК, и их драйверами необходимо перепрограммировать микросхему, точнее, специально выделенную память EEPROM, и редактировать драйвер.

Перепрограммирование микросхемы сводится к изменению кода PID со значения 6001 на любое другое, код VID менять не рекомендуется [4]. Также можно указать имя производителя ЭУ и название устройства. Каждой микросхеме FT232R присваивается уникальный серийный номер – автоматически или заданный пользователем. Наиболее удобно перепрограммировать микросхему с помощью специальной программы MProg с сайта производителя, там же можно найти исчерпывающую справку.

После перепрограммирования микросхемы необходимо редактировать драйвер, изменив код PID в файлах FTD2XXUN.INI и ftd2xx.inf на новый. Для корректной работы множественных FTDI устройств с одним ПК и исключения перезаписи драйверов поверх друг друга рекомендуется изменить имена следующих файлов: FTD2XX.sys, FTD2XX.inf, FTD2XX.dll, FTD2XXUN.ini. Все изменения имен необходимо отразить в файлах FTD2XXUN.INI и ftd2xx.inf, заменив соответствующие. После этих процедур микросхема готова к дальнейшей работе.

## Программная реализация обмена по шине USB с помощью C++

Для работы с USB FTDI устройством можно использовать либо библиотеку функций API [5, 6], либо библиотеку, разработанную специалистами FTDI – второе предпочтительнее. Библиотека FTDI [4] имеет группу функций для чтения и программирования EEPROM, а также обширную группу функций для отслеживания состояния USB устройства, организации и обмена данными. Обмен данными можно производить в синхронном (удобно для одиночных коротких пакетов) и асинхронном (для больших объемов данных) режимах. Скорость обмена выбирается из стандартного ряда от 300 до 921600 бод (бит/сек) либо задается произвольно в соответствии с стандартом USB2.0.

Для организации асинхронного обмена данными в многозадачной операционной системе WindowsXP оптимальным подходом является выделение отдельного потока обмена. Данный поток при поступлении данных генерирует сообщение, обработчик сообщения осуществляет прием данных либо генерирует исключение, например, в случае разрыва линии.

Важно заметить, что файл FTD2XX.lib, входящий в состав драйвера Ftd2xx, совместим с Visual Studio C++ и требует перекомпиляции для BCB (Borland C++ Builder) с помощью утилиты IMPLIB, входящей в состав BCB.

## Заключение

Рассмотренный в статье подход к организации обмена по шине USB – далеко не единственный, но его применение обеспечивает стабильную работу устройств в соответствии со всеми современными стандартами. Организация обмена данными на базе микросхемы FT232R применена в ГРВ- и ИПЧ-оборудовании.

## Литература


1. Колесниченко О.В., Шишигин И.В. Аппаратные средства персональных компьютеров. СПб: БХВ-Петербург, 2005. 1024 с.
2. Тук М. Аппаратные средства IBM PC: Энциклопедия. 2-е изд. СПб: Питер, 2001. 655 с.
3. http://www.usb.org
4. http://www.ftdichip.com
5. Агуров П.В. Практика программирования USB. СПб.: БХВ-Петербург, 2006. 624 с.
6. MSDN for Visual Studio .NET; Copyright 1987–2002 Microsoft Corporation.

# 4 ПЕРСПЕКТИВНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

## ТЕЛЕКОММУНИКАЦИОННАЯ СИСТЕМА ОБРАБОТКИ ПОТОКОВ ДАННЫХ ДЛЯ ТОРГОВО-ПОСРЕДНИЧЕСКИХ ПРЕДПРИЯТИЙ, ЗАНИМАЮЩИХСЯ ВНЕШНЕЭКОНОМИЧЕСКОЙ ДЕЯТЕЛЬНОСТЬЮ

**И.М. Адушкин, О.В. Елисеев**
**Научный руководитель – А.Ю. Мордвинцев (ЗАО «НПК «Промэлектроника»)**

Рассмотрены проблемы современного рынка средств автоматизации бизнес-процессов на предприятиях, ведущих внешнеэкономическую деятельность, выявлены основные недостатки, предложена оригинальная концепция системы обработки потоков данных, объединяющая ряд нестандартных решений.

В настоящее время на рынке делового программного обеспечения становятся все более востребованными системы автоматизации финансово-хозяйственной деятельности. Предприятия постепенно выходят на новый уровень требований к информационному обеспечению служб и подразделений, ориентируются на корпоративные информационные системы. Одной из областей бизнеса, в значительной степени нуждающихся в исследовании возможностей применения и последующем внедрении современных технологий телекоммуникаций, является рынок средств автоматизации внешнеэкономической деятельности (ВЭД) торгово-посреднических предприятий.

Цель данной работы – разработка новой концепции корпоративной информационной системы для предприятия, занимающегося ВЭД, с учетом специфических особенностей данного направления и увеличения роли телекоммуникационных средств в современном бизнесе. В работе предложена новая, отличная от предлагаемых на данный момент, модель системы обработки потоков данных для предприятия, занимающегося внешнеэкономической деятельностью. Применение модели будет способствовать оптимизации затрат трудовых ресурсов торгово-посреднического предприятия, облегчению задачи принятия решений руководством и сотрудниками в процессе выполнения трудовых обязанностей в области внешнеэкономической деятельности.

### Бизнес-процессы организаций, ведущих ВЭД

Деятельность любого торгово-посреднического предприятия, направленную на извлечение прибыли, можно разделить на бизнес-процессы. ВЭД здесь не является исключением. Бизнес-процесс – это последовательность работ, соотнесенная с отдельным видом производственно-хозяйственной деятельности компании и ориентированная на создание новой стоимости, например, выпуск продукции [1]. В упрощенном виде последовательность деятельности предприятия в рассматриваемой области выглядит следующим образом:

1) получаются заявки от заказчика с необходимой сопутствующей информацией;
2) определяются сроки отправки коммерческого предложения партнерам, исходя из определенных факторов (например, таких, как группировка предложений для совместной отправки, загруженность отдела, выходные дни и т.д.);
3) заявки обрабатываются менеджерами по закупкам, результатом работы является информация о ценах и поставщиках по определенным позициям заявки;
4) руководитель отдела, исходя из таких факторов, как конъюнктура внутреннего и внешнего рынка, инфляция, колебания курсов валют, расценки перевозчиков, страховщиков и т.д. вырабатывает определенную степень изменения цены изделия для предоставления ее партнеру в коммерческом предложении;

5) менеджер по внешнеэкономической деятельности формирует коммерческое предложение (список наименований частей оборудования, информация о доступном для поставки количестве, единицах измерения, цене и т.д.) с учетом полученной от менеджеров по закупкам информации. Производится отправка предложения;

6) в случае получения от иностранного заказчика извещения о размещении заказа/выигрыше тендера формируется и подписывается контракт с заказчиком, заключаются договора с поставщиками, производится закупка и экспорт соответствующего оборудования.

В процессе выполнения указанных действий значительное количество сил и времени уходит на механическую обработку данных, часто требуется их повторное внесение. Кроме того, чрезвычайно сложной является работа по отслеживанию фактов проведенной в прошлом работы по определенным видам оборудования, с определенными поставщиками, и вообще по отслеживанию произведенных действий и обработанных данных. Весьма сложным и дорогим образом решается задача телекоммуникаций, являющаяся особо важной при осуществлении международных контактов.

На данный момент на рынке ПО существует некоторое количество стандартных, серийных корпоративных информационных систем, функциональность которых можно в большей или меньшей степени настроить на обеспечение бизнес-процессов ВЭД. Однако все они изначально построены вокруг данных, получаемых из бухгалтерии предприятия, зачастую являющихся недостаточно оперативными. Вследствие невозможности «заточки» функций под данную конкретную область некоторые операции довольно громоздки и неудобны. Кроме того, ни одна из известных подобных систем в достаточной мере не применяет современные средства телекоммуникаций, не обеспечивая достаточной мобильности и гибкости работы сотрудников.

## Краткое описание предлагаемой концепции

Формализованную и генерализованную схему распределения потоков данных на торгово-посредническом предприятии, ведущем ВЭД, в соответствии с бизнес-процессами, описанными выше, а также с учетом необходимости создания общего хранилища данных, в упрощенном виде можно представить в виде диаграммы (рис. 1).

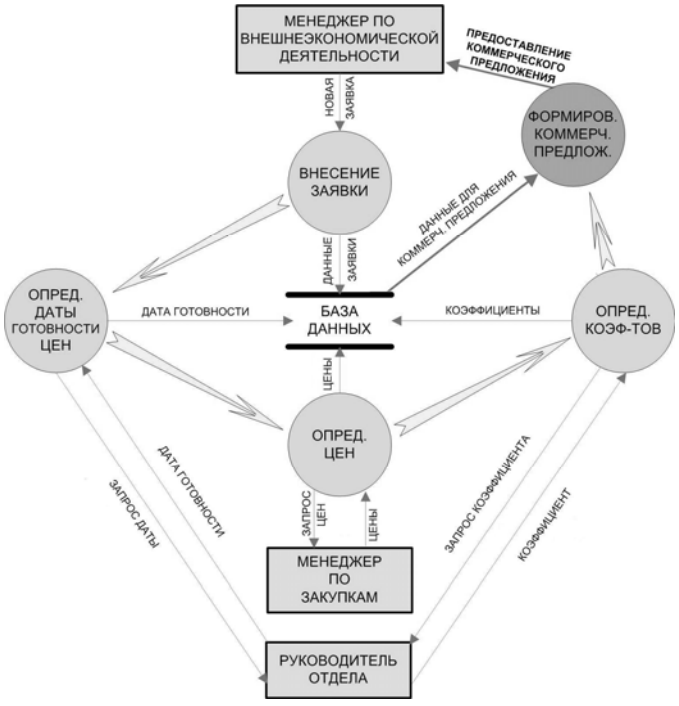



Рис. 1. Диаграмма потоков данных

Основополагающей частью разработанной концепции является автоматизация бизнес-процессов, направленных на реализацию цикла обработки заявок и создания коммерческих предложений. Предоставляя сотрудникам возможность упрощения осуществления каждого из этих процессов, система, кроме того, осуществляет контроль за соблюдением последовательности и своевременности произведенных действий.

Для реализации «общения» с сотрудниками применяется система генерации соответствующих сообщений. Вариант общего алгоритма применения сообщений при реализации бизнес-процессов представлен на рис. 2. Центральная часть системы – база, в которой хранятся все когда-либо обработанные системой потоки данных, а также сообщения, описанные выше. Вариант логической структуры подобной базы данных представлен на рис. 3.

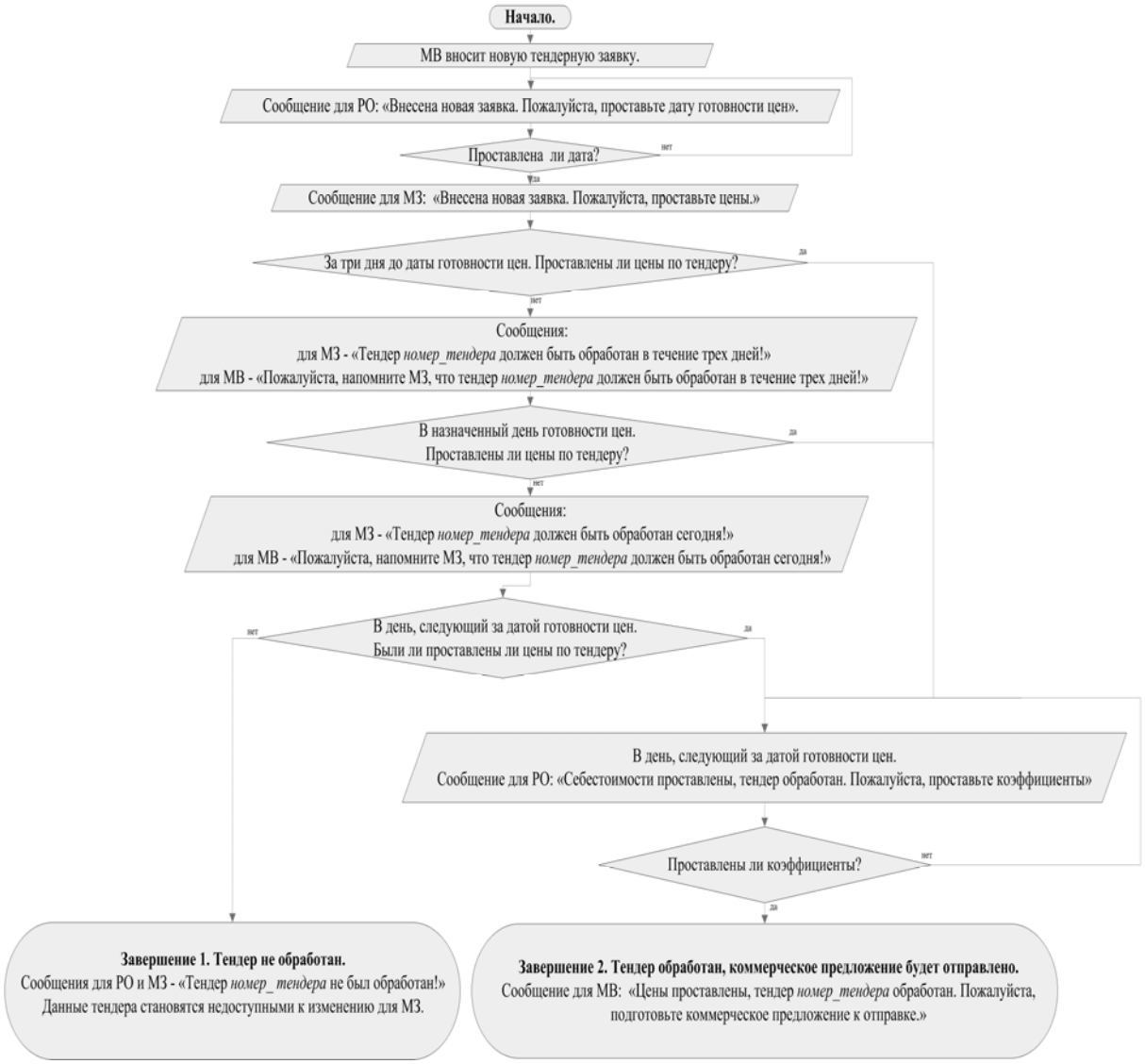



Рис. 2. Алгоритм применения сообщений

С точки зрения технической реализации потоков данных отметим, что абсолютно весь функционал решено перенести на сервер, доступный через веб-интерфейс. Это позволяет избавиться от устаревших для подобной области, с точки зрения авторов, концепций «толстого» и «тонкого» клиента, повсеместно применяемых в корпоративных информационных системах. Единственным необходимым программным инструментом доступа к системе является стандартный веб-браузер. Данный подход серьезным образом облегчает обслуживание системы. Но главное его достоинство – независимость сотрудников от рабочего места, что значительно увеличивает эффективность их работы,

особенно в связи с частыми местными, междугородними и международными командировками. Интерфейс системы должен иметь оптимизацию для работы на карманных ПК, а также предусматривать возможности оптимизации для иных видов мобильных устройств.

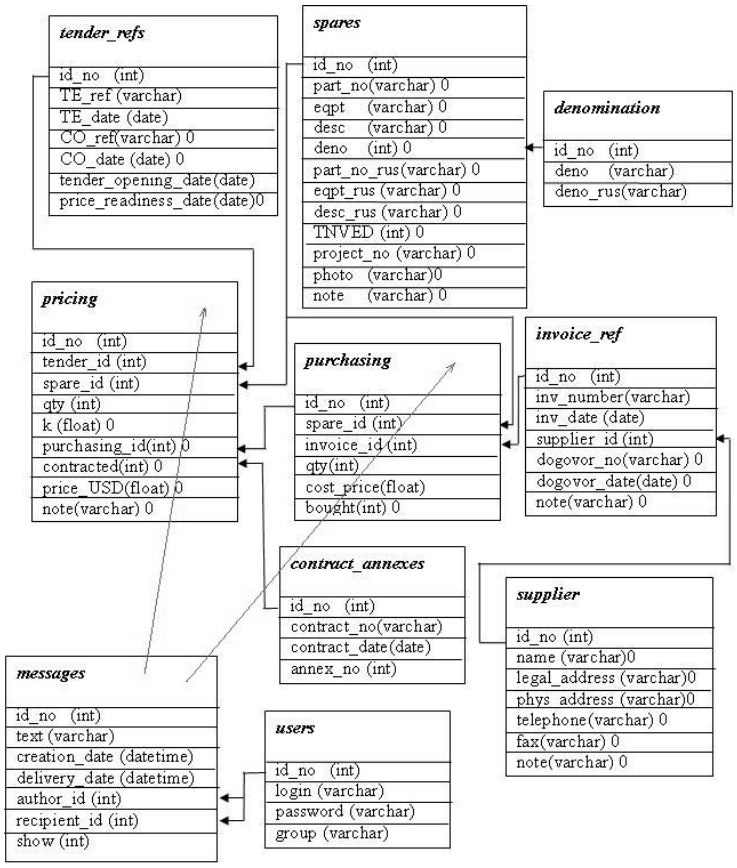



Рис. 3. Структура базы данных

Предлагаемая концепция не имеет целью создание полноценных корпоративных информационных систем. Внешнеэкономическая деятельность зачастую не является единственным видом деятельности предприятия, и иногда возникает необходимость синхронизации потоков данных отдела ВЭД предприятия с процессами, происходящими в других отделах. В связи с этим было принято решение о представлении разрабатываемого класса систем в качестве одного из модулей при формировании корпоративной системы на основе недавно представленной для бизнес-приложений сервис-ориентированной архитектуры (service-oriented architecture, сокр. SOA).

Сервис-ориентированная архитектура представляет собой архитектуру приложений, состоящую из слабо связанных сервисов (например, различные функции, используемые для создания и обработки клиентских заказов) и «потребителей» сервисов (например, пользователи и приложения, которые нужны для создания клиентских заказов) [2]. Модульный дизайн приложений на основе SOA означает, что они могут быть внедрены или усовершенствованы поэтапно с минимальными помехами для конечного пользователя. Напротив, традиционные неделимые приложения должны внедряться целиком, а если часть системы в настоящий момент не используется, то она «выключается». Это усложняет внедрение. С точки зрения разработки приложение на основе SOA отлично подходит для быстрых непрерывных изменений [3].

## Заключение

В результате проделанной работы была разработана оригинальная концепция системы обработки потоков данных для торгово-посреднических предприятий, занимающихся ВЭД, с внедрением нестандартного для сегмента корпоративного ПО решения о распределении вычислительных функций в системе, что позволило повысить гибкость работы сотрудников при условии применения современных средств телекоммуникаций. Разработанная концепция предусматривает выполнение таких задач, как:

- автоматизация обработки больших объемов данных;
- создание базы данных, содержащей необходимую информацию по оборудованию, тендерам, фирмам-поставщикам и выставленным счетам;
- поддержка индивидуальной деятельности сотрудников;
- предоставление сотрудникам справочной информации в удобной форме для поддержки принятия решений, в большей мере соответствующих существующему положению дел;
- возможность реализации удаленного взаимодействия сотрудников отдела;
- проведение автоматизации бизнес-процессов в рамках отдела.

Создан и внедрен прототип разработанного вида систем – проект «Отдел внешнеэкономической деятельности ЗАО НПК Промэлектроника».

## Литература


1. Информационные технологии управления: Учеб. пособие для вузов / Под ред. проф. Г.А. Титоренко. 2-е изд., доп. М.:ЮНИТИ-ДАНА, 2003.
2. Jeremy Westerman. SOA Today: Introduction to Service-Oriented Architecture. Column published in DMReview.com. January 15, 2004.
3. Ден Метьюз. Как выбрать корпоративную систему в 2007 году. РИА ERPNEWS. 26.12.06.
4. Олег Седов. Двойное внедрение. // Intelligent Enterprise. Корпоративные системы. 2002. № 10.
5. Пузанова Е.Н., Бодягин О.В. Внешнеэкономическая деятельность торгово-посреднического предприятия. М: Приор, 1997.
6. Автоматизированные информационные технологии в экономике: Учебник / Под ред. Г.А. Титоренко. М.: Финстатинформ, 1997.
7. http://ssl.ru/ru/ Официальный сайт компании Хостинг-Центр РБК.

# РАЗРАБОТКА ТРЕХУРОВНЕВОЙ АРХИТЕКТУРЫ CMS

**Д.Г. Юдин**
**Научный руководитель – к.т.н., доцент Б.А. Крылов**

В связи с увеличивающейся информатизацией общества услуги по созданию web-сайтов очень востребованы. IT компании не успевают выполнять все поступающие заказы. В таких условиях программисту необходим инструмент для облегчения процесса разработки. Этим инструментом является разработанная CMS.

## Введение

В наше время web-сайты стали непременным атрибутом бизнеса большинства компаний. Непрерывный рост числа пользователей Internet и развитие Internet-технологий делает сеть очень привлекательной для различных коммерческих структур. При их создании преследуются не только рекламные и представительские цели – с помощью web-сайтов компании пытаются сократить расходы, увеличить прибыль, оптимизировать бизнес-процессы.

Для создания сайта компании обращаются к фирме, которая занимается разработкой сайтов. Заказчик, как правило, не очень интересуется, на основе каких технологий будет работать его сайт. Главным для него – чтобы сайт работал постоянно, чтобы было легко наполнять сайт контентом, не привлекая дополнительных работников, чтобы не требовалось сложного обучения, а также невысокая цена.

Для фирмы-разработчика главным является уменьшение срока выполнения работы за большие деньги, уменьшение контактов с заказчиком по поводу уже сделанной работы (обычно осуществляются бесплатные консультации по работе сайта), простота наращивания функционала сайта и, опять же, скорость этой модернизации. Видно, что во многом интересы заказчика и разработчика явно или неявно пересекаются (за исключением стоимости разработки, конечно). В таких условиях разработчику крайне желательно использовать какое-либо специализированное средство для облегчения разработки. Такие программные продукты называют CMS (Content Management System). Список наиболее популярных CMS можно увидеть на сайте http://cmslist.ru/. Казалось бы, достаточно лишь выбрать подходящую. Однако при выборе неизбежно возникают следующие проблемы.

- Бесплатные CMS во многом являются клонами друг друга (часто разветвлениями некогда одного проекта), часто с довольно посредственной функциональностью и гибкостью.
- Коммерческие CMS оказываются слишком дорогими и часто поставляются с закрытым кодом, что делает практически невозможным дополнить функциональность системы.
- Большинство CMS ориентированы на построение сайта из модулей при помощи мыши. Разработчики CMS постоянно стремятся облегчить создание сайта для непрофессионалов, что чаще всего усложняет выполнение нетипичных задач.
- Требуется изучение системы. Может получиться, что выбранная CMS не отвечает запросам, а для того, чтобы перейти на другую или дополнить существующую, требуется время на изучение.

Поэтому фирме-разработчику сайтов желательно иметь собственную CMS, которая отвечает всем запросам, которую при необходимости можно легко доработать, на изучение которой не требуется времени. Впрочем, время тратится на ее разработку.

## Общая концепция

Чтобы определиться с концепцией системы, необходимо рассмотреть, какие компоненты есть в каждом сайте и какие риски по отношению к разработке имеются.

Каждый сайт включает в себя три основные компоненты:

- данные – информация, записанная в определенном формате;
- программа – некая логика обработки данных;
- представление (дизайн).

При создании сайта могут изменяться все три компонента, причем заранее неизвестно, что именно может захотеть поменять заказчик. Поэтому очень желательно иметь возможность изменять независимо любую из трех компонент сайта. К тому же это позволит распределить работу на несколько человек, каждый из которых будет в малой степени зависеть от другого.

Архитектура, при которой осуществляется разделение данных, логики и представления, называется трехуровневой.


Рис. 1. Взаимодействие компонентов в трехуровневой архитектуре

### Независимость данных

Для хранения данных лучше всего использовать базу данных (БД). Это позволит решить ряд проблем, таких как поиск, сортировка, фильтрация. В базе данных проще всего манипулировать данными, и при необходимости можно будет легко изменить их формат и структуру.

Однако использование БД также создает некоторые проблемы. Например, заказчик захочет использовать другую СУБД (Систему управления базой данных) вместо той, на основе которой изначально функционирует CMS. Для этого необходимо предусмотреть слой абстракции программной части от БД. В приведенном выше случае достаточно будет изменить только слой абстракции.

### Независимость логики

Поскольку сначала была рассмотрена независимость данных, то под независимостью логики будем понимать ее независимость от данных, а именно – от формата данных. Понятно, что в БД таблицы могут иметь различное число столбцов, и, соответственно, эти столбцы могут иметь различные имена. Очень важно, чтобы программные модули могли продолжать работать правильно, несмотря на изменение количества столбцов в таблице с данными.

Для обеспечения независимости логики потребуется специальная модель хранения данных. Существует три основных подхода – модульная, объектная и сетевая модели.

**Модульная модель.** В такой модели контент разделен на отдельные модули по типам содержимого. Структура данных зависит от модуля, и вся работа с контентом сосредоточена внутри модуля. Модули независимы и полностью отвечают за работу с документами данного типа. Документы описываются с помощью фиксированного набора характеристик – типы документов строго фиксированы. Расширять функциональность можно за счет добавления нового модуля, замены или редактирования существующего кода. Чаще всего нет никакой системы связей между документами разных мо-

дулей и между документами одного и того же модуля. Стандартный набор типов контента (модулей) таков: ссылки, статьи, файлы, новости, разделы, форум.

Модульная модель достаточно удобна, особенно в случае широко используемых бесплатных или коммерческих системах, когда имеется множество сторонних модулей. Их можно просто добавить в систему. К тому же модульная модель хорошо соотносится с представлениями заказчика о сайте. Однако, данные одного модуля «изолированы» от данных другого. Достаточно часто есть необходимость установить между ними некую взаимосвязь, и тогда приходится вносить достаточно неуклюжие правки в код. Ведь модули могут хранить свои данные в непохожих форматах. Кроме того, нет возможности добавить или удалить поле данных, так как в таком случае придется изменять код модуля (иногда это невозможно).

**Объектная модель.** Объектная модель представления данных оперирует такими понятиями, как класс и объект. Классы определяют структуру данных и представляют собой набор атрибутов (текстовая строка, целое число, изображение и т.д.). Представители класса (объекты) имеют определенную структуру и могут содержать другие объекты, образуя произвольную иерархическую структуру. Объекты могут наследовать свойства, содержание и поведение объектов, которые в них содержатся. Примерами объектов служат документы, картинки, каталоги и учетные записи пользователей. Класс контента не хранит в себе реальных данных – такую информацию содержат объекты (экземпляры класса). Определив один класс, можно создать множество его представителей (контент объектов) [1].

В CMS данные обычно хранятся в реляционной или объектной базе данных. В первом случае объектная модель данных отображается на реляционную модель базы данных. Главное достоинство классов заключается в их абстрактности. Так, класс «Новости» ничем не отличается по своей формату от класса «Фотоаппараты». Атрибуты и их количество могут и будут разными, но для программы главное – формат. Таким образом, для работы с объектами любого класса понадобится один программный компонент. Впрочем, объектная модель не отрицает модульной. Скорее это некое ее расширение. При разработке сайта все равно возникает необходимость в уникальных модулях, в уникальной работе с данными. Однако 90 % всех задач можно решить с помощью одного-единственного программного компонента.

Для работы объектной модели потребуется специальная программа – драйвер объектной модели. Он реализует прозрачный и абстрактный доступ к данным. Он должен поддерживать простые методы получения, добавления и удаления данных, обеспечивать получение информации о классе, а также управлять новыми классами.

**Сетевая модель** представления данных опирается на теорию графов: структура информации представляется в виде узлов с помеченными связями между ними. Фундаментом системы может служить как сетевая, так и традиционная реляционная СУБД, на которую отображена сетевая модель описания данных. В реляционных таблицах хранится информация об узлах, их атрибутах и связях между ними. Связь отличается от атрибута тем, что в ней хранится ссылка на другой узел, а в атрибуте – собственно значение. Для извлечения данных из направленного графа обычно используются рекурсивные процедуры обработки, такие как составление списков узлов, определение атрибутов узла по атрибутам родителя и др.

### Независимость представления

Необходимость изменять оформление сайта возникает постоянно. Некоторые заказчики постоянно выдвигают требования – то подвинуть элемент влево, то вправо, изменить цвет, поменять элементы местами. Если html-код будет перемешан с программным, то такие ситуации причиняют немалый ущерб нервной системе. Править размет-

ку, которая перемешана с кодом, весьма сложно. К тому же просто отдать верстку дизайнеру не получится – он может запросто удалить часть рабочего кода и не заметить, или, если очень «сообразительный», начнет дописывать что-то свое.

Для решения этих проблем существует технология шаблонов. Суть заключается в следующем. Имеется обычный html-документ, в который включены специальные метки. При выполнении программы вместо меток вставляются различные данные. Шаблон может быть один, а может быть несколько, например, по шаблону на модуль. Пример шаблона представлен в листинге.

**Листинг:** Пример шаблона

```
<html>
<head>
<title>:::::{page_title}:::::</title>
</head>
<body bgcolor="{bg_color}">
Добро пожаловать, {user_name}!<br>
</body>
</html>
```

Фигурные скобки имеют специальный смысл при обработке шаблонов – заключенная в них строка интерпретируется как имя переменной, вместо которого подставляется ее значение. Такой шаблон может быть легко отредактирован. Он не содержит лишних строк кода и прост для понимания. И главное, для редактирования представления нет необходимости залезать в программу, достаточно изменить шаблон.

## Используемые языки и программы

В качестве языка разработки выбран язык PHP5, так как на сегодняшний день это самый распространенный язык для разработки web-сайтов. PHP5 является объектно-ориентированным языком, что также весьма удобно для разработки. Для работы системы используется сервер под управлением FreeBSD, на котором установлены www-сервер Apache и СУБД MySQL.

## Архитектура системы

Итак, система должна быть основана на трех компонентах: драйвер БД, драйвер объектной модели и драйвера шаблонов.

**Драйвер БД** должен выполнять следующие функции:

- подключение к БД;
- запрос данных из БД;
- выдача результата запроса к БД;
- обработка ошибок при запросах к БД.

Драйвер БД является классом со стандартизированным в рамках разработанной CMS API. Поэтому он может быть легко заменен при переходе на другую СУБД. В работе системы создается один первый экземпляр класса, который устанавливает соединение с БД. Далее можно делать копии данного объекта для работы через одно и то же соединение. При этом результаты запросов через каждую копию объекта хранятся раздельно.

**Драйвер объектной модели.** В работе объектной модели используются три таблицы в БД и один программный компонент. Таблицы в БД отвечают за следующие функции: хранение описания классов, хранение описания атрибутов, хранение связи классов и атрибутов, а также данных о характере связи (рис. 2).

Классы имеют различные типы данных и типы привязки в системе. Типы данных следующие: «Объект», «Дерево», «Форма», «Контейнер». Объект – это класс, который

хранит обычные табличные данные. Дерево – класс, который хранит в себе структуру типа «дерево». Форма не хранит в себе данных, этот класс используется для автоматического создания форм. Контейнер – класс, включающий один экземпляр. Классы типа «контейнер» используются для хранения каких-либо настроек. Типы привязки – «Системный», «Проектный», «Дистрибутивный» – созданы для автоматической генерации чистого дистрибутива системы.

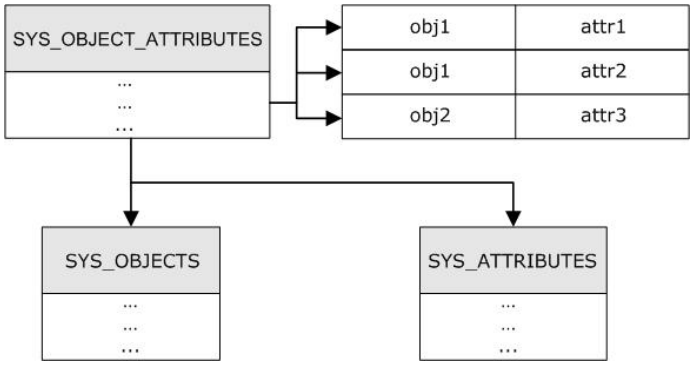



Рис. 2. Схема таблиц, отвечающих за объектную модель

**Драйвер шаблонов** выполняет функции компоновки и вывода результатов пользователю. Процесс обработки шаблона можно разделить на четыре этапа:

- регистрация файлов – регистрация всех файлов, обрабатываемых сценариями шаблонов;
- регистрация переменных – регистрация всех переменных, которые должны заменяться своими значениями в зарегистрированных файлах;
- обработка файлов – замена всех переменных, находящихся между ограничителями, в зарегистрированных файлах;
- вывод файла – вывод обработанных зарегистрированных файлов в браузере.

Для совместимости со всеми хостингами шаблонизатор целиком написан на PHP. Для обозначения мест вставки данных используются имена в фигурных скобках. Такая схема наиболее просто для понимания. Форматирование же значений (строчные/прописные буквы, длинный/короткий формат даты) обеспечивается с помощью описания атрибута соответствующего формата.

В итоге имеем три независимых компонента с разработанными API для взаимодействия. На основе этих компонентов была создана CMS, которая называется ExpertCMS. Данная система применяется во многих проектах.

## Заключение

При разработке решалась задача упрощения разработки web-сайтов не за счет автоматической генерации сценариев, а за счет обеспечения проекта удобной архитектурой, которая была реализована на основе трех программных компонентов. Данная реализация позволяет в случае необходимости безболезненно изменить любой из компонентов и совместима со всеми хостинг-платформами. Совместимость достигается за счет того, что все компоненты реализованы с использованием стандартных конструкций языка разработки. Польза данной архитектуры подтверждена успешным внедрением в коммерческих проектах.

# СРАВНИТЕЛЬНЫЙ АНАЛИЗ МОДЕЛЕЙ УЧЕТА ПОПРАВОК НА РАСПРОСТРАНЕНИЕ СИГНАЛОВ КОСМИЧЕСКИХ АППАРАТОВ В ТРОПОСФЕРЕ

**А.С. Бандура, А.А. Скобелин**
**Научный руководитель – д.т.н., профессор, В.Л. Ткалич**

Статья посвящена сравнительному анализу моделей учета поправки на распространение сигналов космических аппаратов в тропосфере. В работе произведен анализ влияния условий наблюдения космических аппаратов на точность определения поправки на основании результатов работы составленной программы.

## Введение

Спутниковые технологии высокоточного сличения времени по сигналам космических навигационных систем (КНС) ГЛОНАСС и GPS за последние десятилетия достигли значительного прогресса.

Использование сигналов космических навигационных систем ГЛОНАСС и GPS для передачи времени в последние десятилетия приобрело неоспоримое значение и стало основным способом сличений часов различных лабораторий под эгидой Международного бюро мер и весов (BIPM), на основании которых формируются мировые системы исчисления шкал атомного TAI и координированного UTC времени.

Точность передачи системного времени КНС достигает единиц наносекунд, и прослеживается тенденция к дальнейшему повышению точности передаваемых сигналов времени.

Решение широкого круга задач, решаемых с помощью КНС невозможно без обеспечения привязки шкал времени аппаратуры, участвующей в вычислениях, причем недостаточная точность привязки может резко понизить точность решения этих задач. Одним из основных требований является обеспечение привязки шкал времени (ШВ) космического аппарата (КА) и приемника к ШВ системы.

## 1. Постановка задачи

### 1.1 Общая методика решения временной задачи

Общую методику определения расхождения БШВ КА относительно ШВ приемника, в соответствии с которой происходит решение задачи, можно представить в виде выражения:

$$\Delta\widehat{T}_{ПРМ-КА} = S^i - T_{ГЕОМ} - \widehat{\tau}_{ион}{}^i - \widehat{\tau}_{троп}{}^i - \widehat{\tau}_{прив}{}^i - \widehat{\tau}_{рел}{}^i - \widehat{\tau}_{прм}{}^i, \qquad (1.1)$$

где $S^i$ – измеренная псевдодальность между фазовым центром передающей антенны *i*-го КА и фазовым центром приемной антенны приемника. Измерение псевдодальности есть измерение расхождения сигнала 1 Гц ШВ местного эталона времени и частоты (ЭВЧ) относительно сигнала 1 Гц, принимаемого с КА, выраженное в секундах;

$$T_{ГЕОМ} = \frac{D^i}{c}, \qquad (1.2)$$

где $D^i$ – геометрическая дальность от КА до измерительного пункта,

$$D^i = \sqrt{[X_П - X_{КА}(t_K)]^2 + [Y_П - Y_{КА}(t_K)]^2 + [Z_П - Z_{КА}(t_K)]^2},$$

$X_П$, $Y_П$, $Z_П$ – прямоугольные геоцентрические координаты приемника в системе координат WGS-90; $X_{КА}(t_K), Y_{КА}(t_K), Z_{КА}(t_K)$ – прямоугольные геоцентрические координаты *i*-го КА в системе координат на момент времени $t_K$; $c$ – скорость света; $\widehat{\tau}_{ион}{}^i$ – временная задержка сигнала, обусловленная влиянием ионосферы; $\widehat{\tau}_{троп}{}^i$ – временная задержка сигнала, обусловленная влиянием тропосферы; $\widehat{\tau}_{прив}{}^i$ – временная поправка, обусловленная

тем, что псевдодальность измеряется от фазового центра антенны КА ГЛОНАСС, а соответствующие этому измерению эфемериды привязаны к центру масс КА; $€_{рел}{}^{i}$ – временная задержка, обусловленная влиянием релятивистских эффектов взаимного движения ШВ; $€_{прм}{}^{i}$ – временная задержка в приемнике и в соединительных кабелях между приемной антенной и приемником [2].

### 1.2 Учет поправки на распространение сигнала КА в тропосфере

Задержка радиосигнала в атмосфере обусловлена искривлением траектории распространения радиоволн – рефракцией, вызванной неоднородным по высоте распределением диэлектрической проницаемости. Наиболее широко распространено представление тропосферной задержки в виде сухой и влажной составляющих:

$$\hat{\tau}_{mpon} = \tau_{dry} + \tau_{wet}, \qquad (1.3)$$

где $\tau_{dry}$ – «сухая» составляющая, обусловленная изменением индекса рефракции, происходящем под влиянием изменения плотности воздуха под влиянием гравитации Земли; $\tau_{wet}$ – «влажная» составляющая, обусловленная концентрацией водяных паров в нижних слоях атмосферы.

Поправка на распространение сигнала КА, находящегося в зените, пересчитывается для общего случая, когда КА находится на угле возвышения менее 90°, вдоль наклонной линии распространения с помощью так называемой Mapping-функции, простейший вид которой может быть представлен в виде соотношения

$$m(e) = \frac{1}{\sin e}, \qquad (1.4)$$

где $m(e)$ – значение Mapping-функции; $e$ – угол возвышения КА.

Значение Mapping-функции колеблется от 1 в зените до приблизительно 6 при угле возвышения 15°.

Таким образом, полное значение $\hat{\tau}_{mpon}$ может быть выражено соотношением

$$\hat{\tau}_{mpon} = \tau_{dry} \times m_{dry}(e) + \tau_{wet} \times m_{wet}(e).$$

В данной работе рассмотрены модель Блэка и модель Хопфилда.[3]

**Модель Блэка** предполагает следующие соотношения для «сухой» и «влажной» составляющих и Mapping-функции:

$$\tau_{dry} = 2.343 \times p \times \frac{T - 4.12}{T} \times m(h_d, e), \qquad (1.5)$$

$$\tau_{wet} = k_w \times m(h_e, e), \qquad (1.6)$$

где p – атмосферное давление; T – температура; H – высота тропосферы в зените; $k_w$ – коэффициент.

Сухая составляющая определяется температурой и давлением, в то время как влажная – широтой места и временем года. Например, для средних широт весной $k_w$=0.2.

$$m(h,e) = \frac{1}{\sqrt{1 - \dfrac{\cos e}{1 + (1 - l_s) \times \dfrac{h}{r_s}}}}, \qquad (1.7)$$

где $l_s$=0.85 – эмпирический коэффициент (скалярный фактор); $r_s$ – расстояние от центра Земли до приемника; $h$ – высота тропосферы над приемником.

Согласно модели Блэка, высота тропосферы для «влажной» составляющей может быть признана константой и равна ~ 13 000 м, а для «сухой» составляющей моделируется в виде [3,4]

$$h_d = 148.98(T - 4.12). \quad (1.8)$$

**Модель Хопфилда** предполагает следующие соотношения для «сухой» и «влажной» составляющих и Mapping-функции:

$$\tau_{dry} = \int_0^{h_d} N_d dh = N_{0d} \frac{h_d}{\mu + 1}, \quad (1.9)$$

$$\tau_{wet} = \int_0^{h_w} N_w dh = N_{0w} \frac{h_w}{\mu + 1}, \quad (1.10)$$

где $N$ – преломляющая способность атмосферы, зависящая от давления, влажности и температуры;

$\mu = (g/R\alpha) - 1$,

g – ускорение свободного падения в точке расположения приемника; R – универсальная газовая постоянная; α – вертикальный градиент температуры; $h_i$ – высота тропосферы над приемником.

По модели Хопфилда [3,4]

$$h_d = 40.209 + 0.154 \times T_s; \quad (1.11)$$

$$h_w = 10.474 + 0.111 \times T_s. \quad (1.12)$$

## 2. Результаты расчета

В работе рассмотрено применение различных моделей учета тропосферной поправки при решении временной задачи. Расчетная программа, используемая при получении результатов, написана в среде программирования Delphi 7.

Сравнительный анализ моделей в работе проводился с использованием данных двухчастотного GPS приемника, установленного на контрольно-измерительном пункте системы GPS Брюсселе. Использованы измерения на временном интервале в одни сутки.

Производилось решение временной задачи по одному КА системы GPS. При этом учет тропосферной поправки производился по разным моделям, затем на основании анализа результатов расчетов делался вывод о точности учета поправки с использованием той или иной модели.

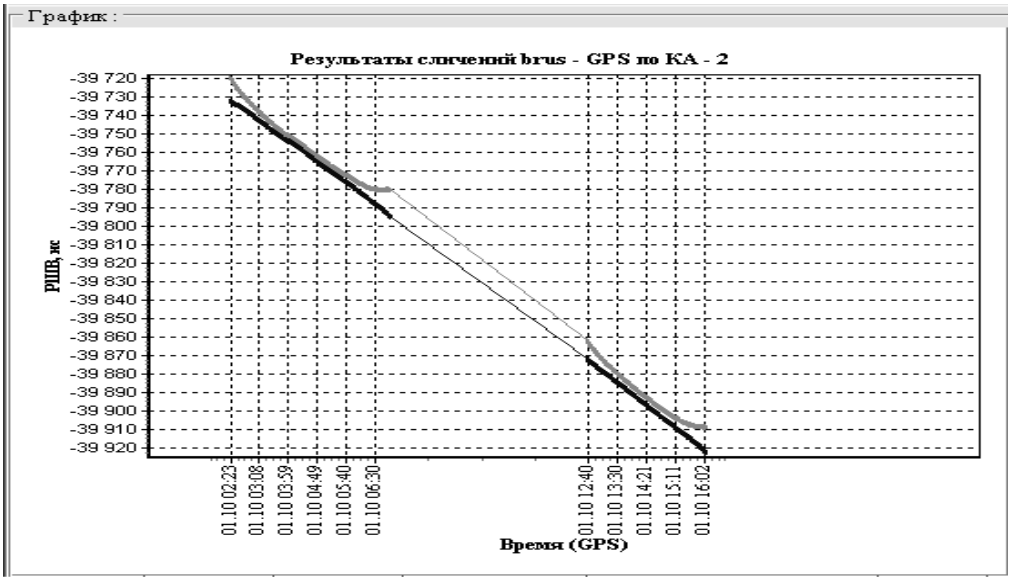



Рис. 1. Расчет РШВ ЭЧВ-БШВ КА при использовании моделей Хопфилда (черный цвет) и Блэка (серый цвет)

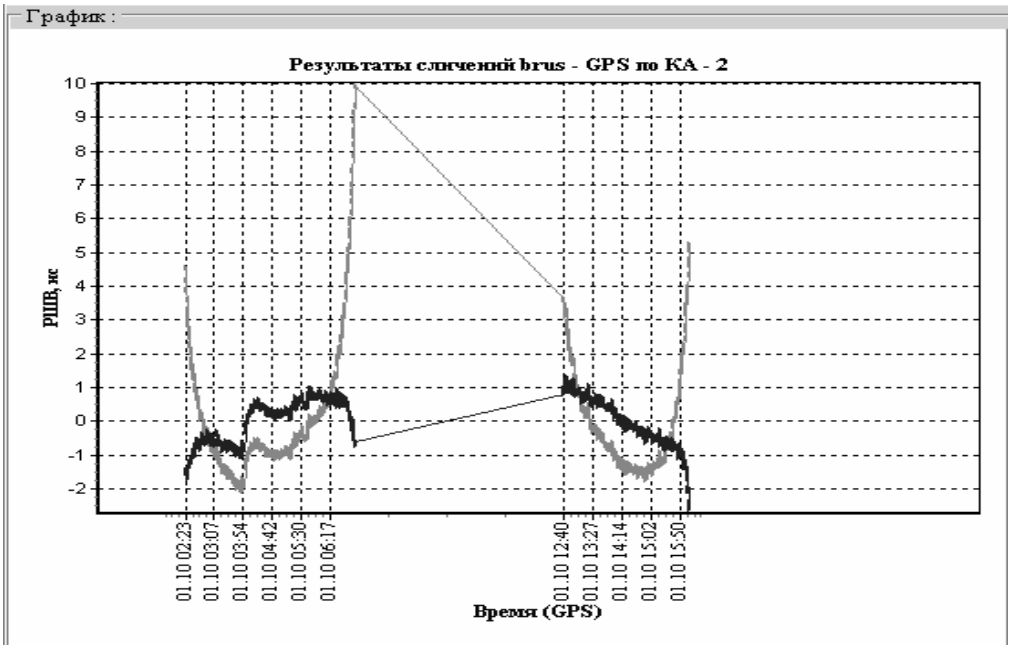



Рис. 2. Расчет РШВ ЭЧВ-БШВ КА при использовании Хопфилда (черный цвет) и Блэка (серый цвет) после вычитания линейного тренда

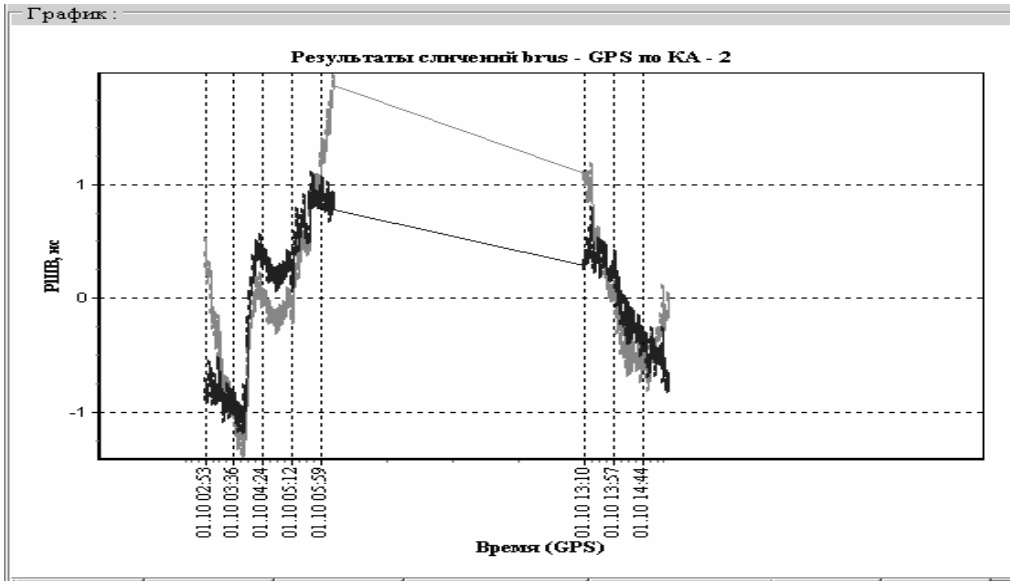



Рис. 3. Расчет РШВ ЭЧВ-БШВ КА при использовании Хопфилда (черный цвет) и Блэка (серый цвет) после вычета линейного тренда при углах возвышения более 20°

Исходя из того, что бортовой эталон КА имеет на порядок худшую стабильность частоты, чем наземный эталон времени и частоты, можно сделать вывод о линейной зависимости от времени значения расхождения шкал времени (РШВ) этих эталонов. На рис. 1 хорошо виден линейный характер величины РШВ эталона приемного пункта относительно часов КА PRN 2.

Если усреднить результаты определения РШВ по линейной модели и затем вычесть полученную функцию усреднения, то становятся хорошо видны ошибки, вносимые использованием обеих моделей. Видно, что ошибки при расчетах по модели Блэка почти на порядок больше. Однако, если исключить из расчета измерения при низких (ниже 20°) углах возвышения, то становится видно, что вносимые обеими моделями ошибки примерно одинаковой величины.

Следует также отметить, что при расчетах я не располагал реальными данными о температуре, влажности и атмосферном давлении, поэтому в модели подставлялись сред-

ние значения этих параметров: температура 20 °С, давление 760 мм рт. ст., относительная влажность воздуха 50 %.

## Заключение

На основании сравнительного анализа можно сделать вывод о том, что на относительно высоких углах возвышения обе рассмотренные модели позволяют достаточно точно учесть задержку распространения сигнала в тропосфере. Однако на углах возвышения ниже 20 % ошибки при использовании модели Блэка составляют до 10–15 нс, что недопустимо при необходимости решения навигационно-временной задачи с высокой точностью.

Среди путей дальнейшего повышения точности учета тропосферной задержки следует отметить применение реальных метеорологических данных.

## Литература


1. Под ред. Харисова В.Н., Перова А.И., Болдина В.А. Глобальная спутниковая навигационная система ГЛОНАСС. М.: ИПРЖР, 1999.
2. Соловьев Ю.А. Системы спутниковой навигации. М.: Эко-Трейдз, 2000.
3. Korak Saha, Suresh Raju and K. Parameswaran. Neutral Atmospheric Refraction on Microwave Propagation and Its Implication on GPS Based Ranging System., 2003.
4. Baker H.C., Dodson A.H., Jerett D. and Offlier D. (1998) Ground-based GPS Water Vapour Estimation for Meteorological Forecasting. // Procs IX Conference on Satellite Meteorology and Oceanography, American Meteorological Society. P. 523–526.

# МЕТОДЫ ИНТЕГРАЦИИ ПРИЛОЖЕНИЙ

**А.Ю. Иваненчук, А.А. Малинин**
**Научный руководитель – доцент Н.Ю. Иванова**

При разработке и внедрении информационных систем возникает задача их интеграции как в уже сложившуюся информационную среду предприятия, так и в среду окружающего экономического сообщества. В статье рассматриваются современные методы интеграции приложений и используемые при этом технологии. Результатом работы является обзор, выполненный в процессе работы, а также ряд рекомендаций для IT-специалистов, стремящимся эффективно управлять интеграцией в процессе разработки и внедрении корпоративных информационных систем.

При разработке и внедрении информационных систем возникает задача их интеграции как в уже сложившуюся информационную среду предприятия, так и в среду окружающего экономического сообщества. Несмотря на то, что задача интеграции корпоративных приложений не нова, она по-прежнему остается одной из серьезнейших задач, с которыми время от времени приходится сталкиваться многим компаниям.

Интеграция приложений – это сложная и многогранная задача, которая охватывает все уровни корпоративной системы – ее архитектуру, аппаратное и программное обеспечение и процессы.

При интеграции бизнес-процессов компания должна определять, реализовывать и управлять процессами обмена корпоративной информацией между различными бизнес-системами. Благодаря этому организация может упростить операции, сократить расходы и улучшить реагирование на запросы клиентов. Комплексная интеграция включает управление процессами, моделирование процессов и технологический процесс, который охватывает различные задачи, процедуры, архитектуры, требуемую входную и выходную информацию, а также средства, необходимые для каждого шага в бизнес-процессе.

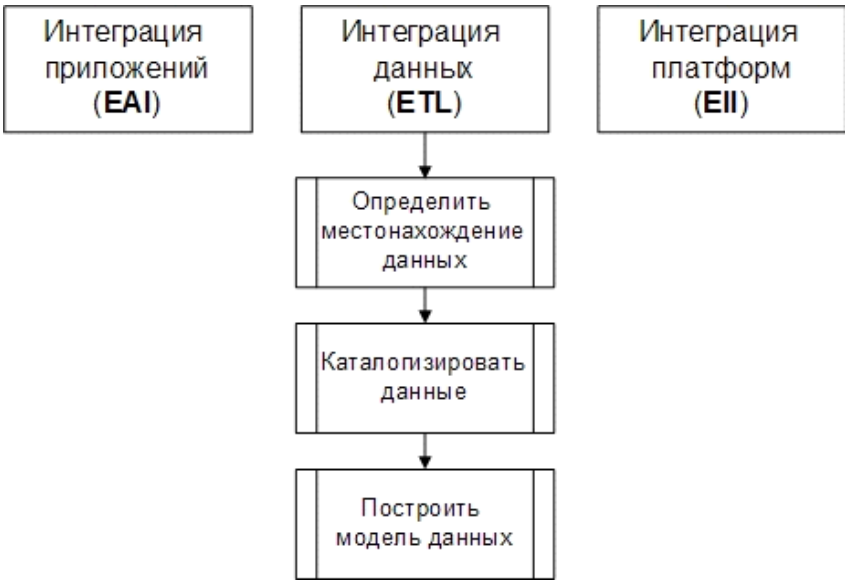



Рис. Методы интеграции

**Интеграция приложений.** На этом уровне интеграции целью является объединение данных или функции одного приложения с другим, благодаря чему обеспечивается интеграция, близкая к реальному времени. Интеграция приложений используется – и это далеко не полный список – для интеграции B2B, внедрения CRM-систем, которые интегрированы с корпоративными серверными приложениями, web-интеграции и построения web-сайтов, которые поддерживают многочисленные бизнес системы. Кроме того, может

потребоваться проведение специальной интеграции, особенно когда требуется интегрировать существующее приложение с вновь устанавливаемым приложением.

**Интеграция данных.** Залогом успешной интеграции приложений и бизнес-процессов является интеграция данных и систем баз данных. Прежде чем приступать к интеграции, необходимо идентифицировать (определить местонахождение) и каталогизировать данные, построить модель данных. По завершении этих трех шагов данные можно совместно использовать/распространять в системах баз данных.

**Интеграция платформ.** Чтобы завершить интеграцию систем – базовой архитектуры, аппаратного и программного обеспечения – необходимо интегрировать разнесенные части гетерогенной сети. Интеграция платформ касается процессов и инструментов, с помощью которых эти системы могут осуществлять безопасный и оптимальный обмен информацией. В результате данные могут беспрепятственно передаваться по различным приложениям. Например, определение того, как нужно надежно передавать информацию с NT- на UNIX-машину, является чрезвычайно непростой задачей по интеграции всей корпоративной системы.

В технологиях интеграции принята следующая терминология: интеграция корпоративных приложений (enterprise application integration, сокр. EAI), интеграция корпоративной информации (enterprise information integration, сокр. EII) и программное обеспечение для извлечения, преобразования и загрузки данных (extract, transform and load, сокр. ETL).

**EAI** – это технология, с помощью которой организация добивается централизации и оптимизации интеграции корпоративных приложений, обычно используя те или иные формы технологии оперативной доставки информации (push technology), которая управляется внешними событиями (event-driven).

**ETL** – это технология, которая преобразует данные (обычно с помощью их пакетной обработки) из операционной среды, включающей гетерогенные технологии, в интегрированные, согласующиеся между собой данные, пригодные для использования в процессе поддержки принятия решений.

**EII** – это технология для интеграции в режиме реального времени несопоставимых типов данных из многочисленных источников как внутри, так и за пределами корпорации. Инструменты EII обеспечивают универсальный уровень доступа к данным и используют технологию поиска информации (pull technology) или возможности работы по запросам.

Как известно, при решении большинства интеграционных задач данные должны быть преобразованы тем или иным способом – структурно (например, чтобы снять различия между исходной и целевой схемами для обеспечения согласованности данных) или семантически (например, чтобы устранить несоответствия в бизнес-значениях в различных системах). Технологии интеграции данных могут существенно отличаться друг от друга с точки зрения возможностей преобразования – от незначительной поддержки трансформирования или ее отсутствия (в случае использования таких технологий, как передача файлов) до широких возможностей преобразования (например, средства ETL).

В завершение свежует отметить, что IT-отделам, стремящимся эффективно управлять интеграцией, следует останавливаться на использовании стандартных технологиях, которые могут быть использованы во всей организации. Следует оценивать инструменты с точки зрения информационной архитектуры. А поскольку для решения задач интеграции данных нет универсального подхода, то необходимо соотносить требования, предъявляемые к интеграционному решению, с реальными характеристиками имеющихся технологий.

# АЛГОРИТМ УПРАВЛЕНИЯ ЧАСТОТОЙ ОПОРНОГО ГЕНЕРАТОРА ДЛЯ СИСТЕМЫ МОНИТОРИНГА СРЕДСТВ СИНХРОНИЗАЦИИ

**А.А. Скобелин, А.С. Бандура**
**Научный руководитель – д.т.н., профессор В.Л. Ткалич**

В статье рассматривается схема системы мониторинга частотно-временной информации, передаваемой импульсно-фазовыми радионавигационными станциями системы единого времени, приводится алгоритм управления частотой опорного генератора измерительной системы, анализируются полученные результаты.

## Введение

Под мониторингом понимается наблюдение и контроль над каким-либо процессом. В данном случае задачами мониторинга являются: измерение мгновенных привязок принимаемых меток времени относительно выбранного эталона, ведение архива измерений с целью определения зависимостей ухода шкал времени, формируемым по внешним меткам времени, прогнозирование дальнейшего ухода шкал времени с целью внесения дополнительных поправок для потребителей.

Измерение расхождения двух шкал времени [1] (привязка) и оценка стабильности шкалы времени, формируемой по внешним меткам времени, становится возможным только при том условии, что стабильность собственной шкалы времени измеряющей системы (оцениваемая, как правило, через среднее квадратическое отклонение), превосходит стабильность принимаемой шкалы.

В настоящее время для решения задачи мониторинга используются измерения привязок принимаемой шкалы времени относительно шкалы времени, формируемой по сигналам космических навигационных систем. Этот метод обладает существенным недостатком: при пропадании навигационного сигнала либо ухудшении точности по различным причинам измерения становятся невозможными.

Предлагаемый метод основан на использовании хранителей шкалы времени – высокостабильных генераторов, обеспечивающих заданную точность за период измерения. В работе приводится схема измерительного комплекса, алгоритм управления частотой опорного генератора и анализируются полученные результаты.

## Структурная схема измерителя

Измерительная система построена на основе двух приемников сигналов системы единого времени: приемо-измерительного модуля сигналов космических навигационных систем (КНС) ГЛОНАСС и GPS и приемника сигналов импульсно-фазовых радионавигационных станций СДВ-диапазона

Приемоизмеритель КНС формирует метку времени частотой 1 Гц и код оцифровки принимаемой ШВ. Эти сигналы принимаются устройством синхронизации стандарта частоты, которое решает одновременно две задачи – установку собственной ШВ по сигналам КНС и установку номинала частоты опорного генератора для обеспечения минимальной начальной скорости ухода шкалы времени при переходе в режим автономного хранения (при пропадании сигналов средства привязки).

Модуль приема сигналов формирует метку времени 1 Гц, которая поступает на измеритель временных интервалов, служащий для измерения расхождения двух меток времени. Структурная схема предлагаемой модели приведена на рис. 1.

В основе предлагаемого метода синхронизации шкалы времени по сигналам КНС лежит следящая система фазовой автоподстройки частоты опорного генератора с перерегулированием, которая, помимо установки номинала частоты генератора, позволяет

плавно, без скачков, сдвигать формируемую шкалу времени. Сигнал частотой 5 МГц со стандарта частоты поступает на устройство синхронизации стандарта частоты, где он делится до частоты 1 Гц и измеряется расхождение между ним и меткой времени КНС. Полученные мгновенные результаты привязки формируемой шкалы времени непрерывно комплексируются и обрабатываются по методу наименьших квадратов [2], в результате чего получается оценка действительного значения частоты (ДЗЧ) опорного генератора и положения шкалы времени. В конце интервала обработки рассчитывается воздействие на опорный генератор, устраняющее отклонение ДЗЧ относительно номинала, а также некая дополнительная отстройка частоты, позволяющая свести шкалу времени к шкале UTC(SU) в течение следующего интервала. Код ШВ используется для разрешения неоднозначности измерений.

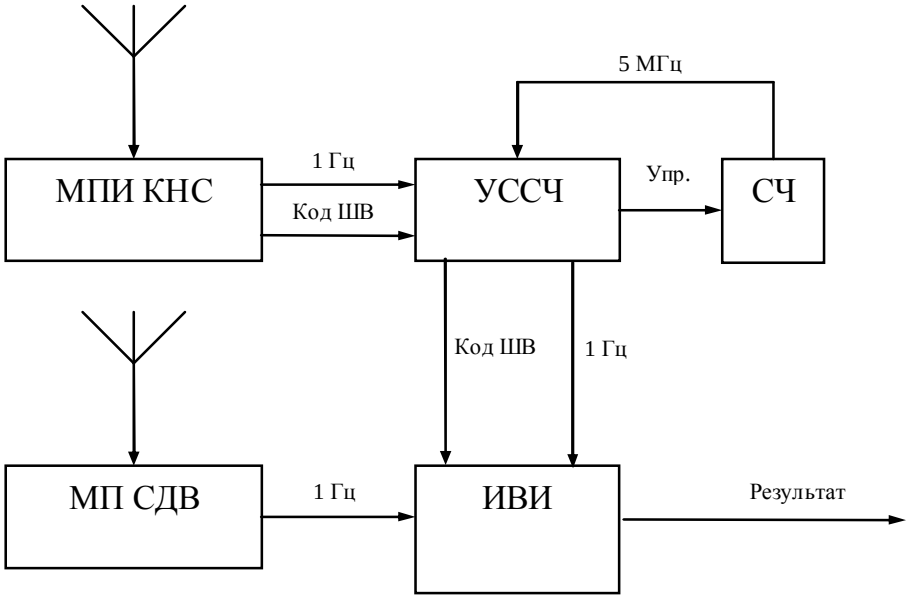



Рис. 1. Структурная схема измерительной системы: МПИ КНС – модуль приемо-измерительный КНС, МП СДВ – модуль приемный сигналов ИФРНС СДВ–диапазона, ИВИ – измеритель временных интервалов, УССЧ – устройство синхронизации стандарта частоты, СЧ – стандарт частоты, Упр. – управляющее воздействие

Измерение временных интервалов между метками времени производится путем заполнения измеряемого интервала сигналом высокой частоты (~125 МГц) и измерения количества импульсов, что дает среднюю ошибку измерения (с учетом температурных флуктуаций частоты) ~10 нс.

### Алгоритм управления частотой опорного генератора

Блок–схема алгоритма управления частотой опорного генератора приведена на рис. 2. Алгоритм отрабатывает один раз в секунду (по приходу метки времени собственной ШВ). Под фазированием понимается грубая установка ШВ с минимальным значением перестройки 200 нс. Длительность одного интервала обработки выбрана, исходя из характеристик опорного генератора, приведенных ниже, и составляет 6 ч. (21600 с). Основные характеристики опорного генератора (рубидиевый стандарт частоты) приведены в таблице.

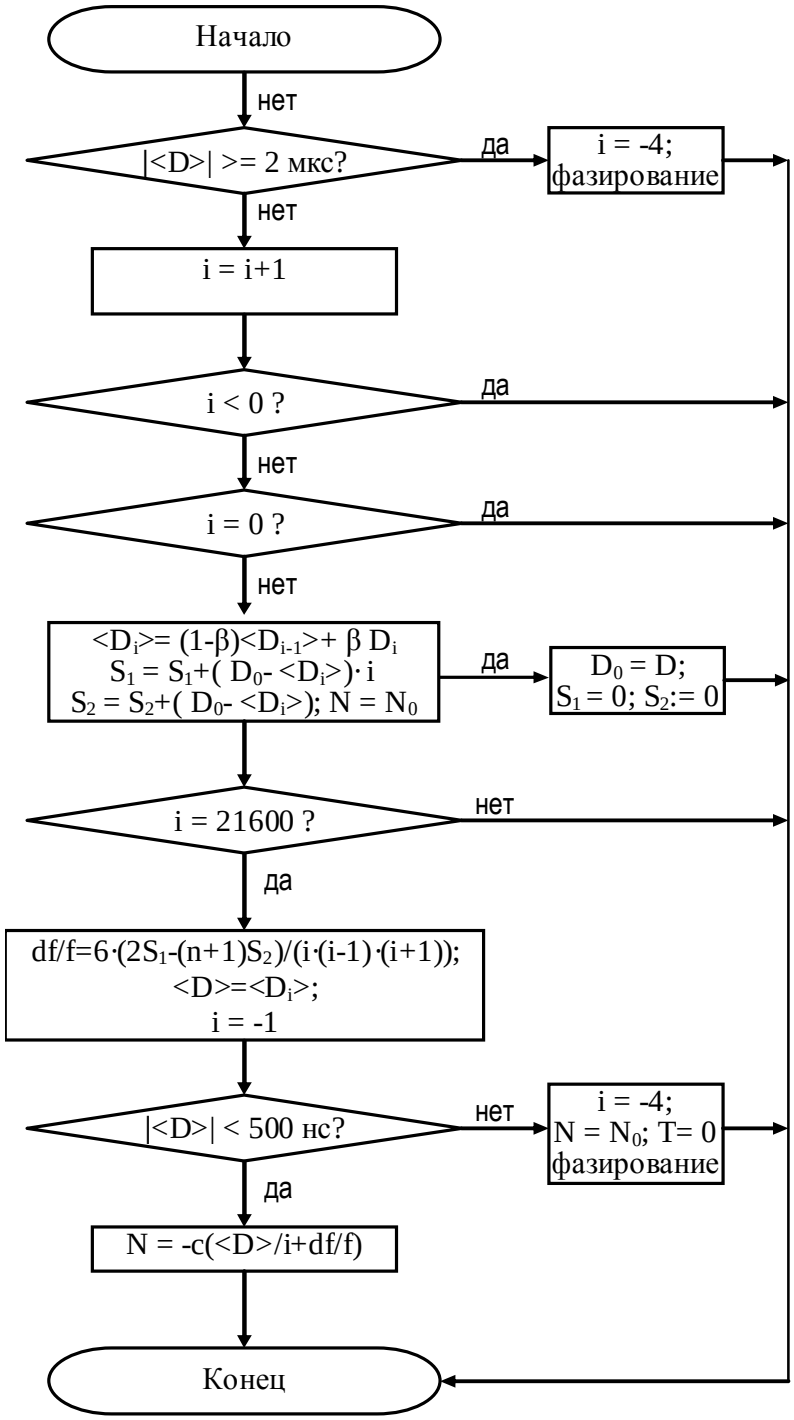



Рис. 2. Блок-схема алгоритма управления частотой опорного генератора: df/f – относительное отклонение ДЗЧ от номинального значения частоты 5 МГц; N –текущий код, установленный в синтезаторе частоты опорного генератора; $N_0$ – расчетное значение кода синтезатора частоты, при котором df/f минимально; i – счетчик циклов внутри интервала обработки (21600 с); D – результат измерения расхождения формируемой метки времени от метки времени приемоизмерителя КНС; $D_0$ – результат измерения расхождения в начале интервала обработки; $D_i$ – результат измерения расхождения в текущем цикле интервала обработки; <D> – сглаженная оценка расхождения (для устранения флуктуаций метки времени КНС); $S_1$, $S_2$ – промежуточные расчетные величины; c – величина, обратная минимальному значению перестройки стандарта частоты; β – постоянная сглаживания <D>

| Время измерения и время наблюдения | Средняя квадратическая относительная вариация частоты, не более |
|---|---|
| $\tau_и$ = 1 с, $\tau_н$ = 100 с | $2 \cdot 10^{-11}$ |
| $\tau_и$ = 1000 с, $\tau_н$ = 6 ч | $3 \cdot 10^{-12}$ |

Таблица. Характеристики опорного генератора

**Результаты эксперимента**

В работе оценивается эффективность предложенного алгоритма управления частотой опорного генератора. Программа, реализующая указанный алгоритм, написана на ассемблере микроконтроллеров семейства MCS-51 [3]. Результаты измерения, обработанные с помощью ПЭВМ, приведены на рис. 3–6.

Из графика, представленного на рис. 3, видно, что алгоритм полностью отработал два раза. За этот период было сформировано два управляющих воздействия на опорный генератор (рис. 6). После первого воздействия (при достижении значения около 500 нс) расхождение стало практически линейно возрастать, второе же воздействие уменьшило скорость ухода частоты и привело график расхождения в почти горизонтальное положение. Это говорит об эффективности указанного алгоритма и позволяет сделать вывод о возможности его применения в системе мониторинга сигналов средств синхронизации.

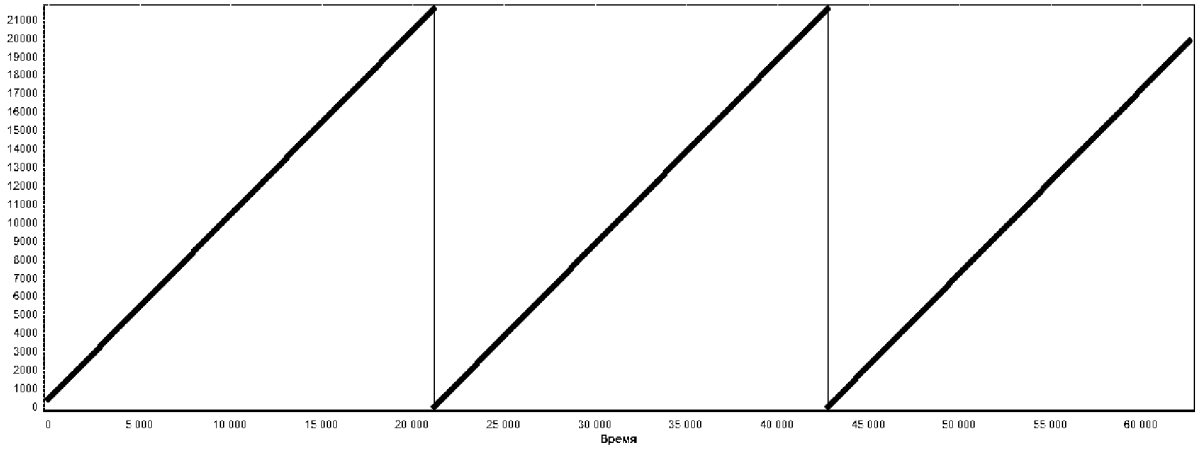



Рис. 3. График зависимости параметра i от времени

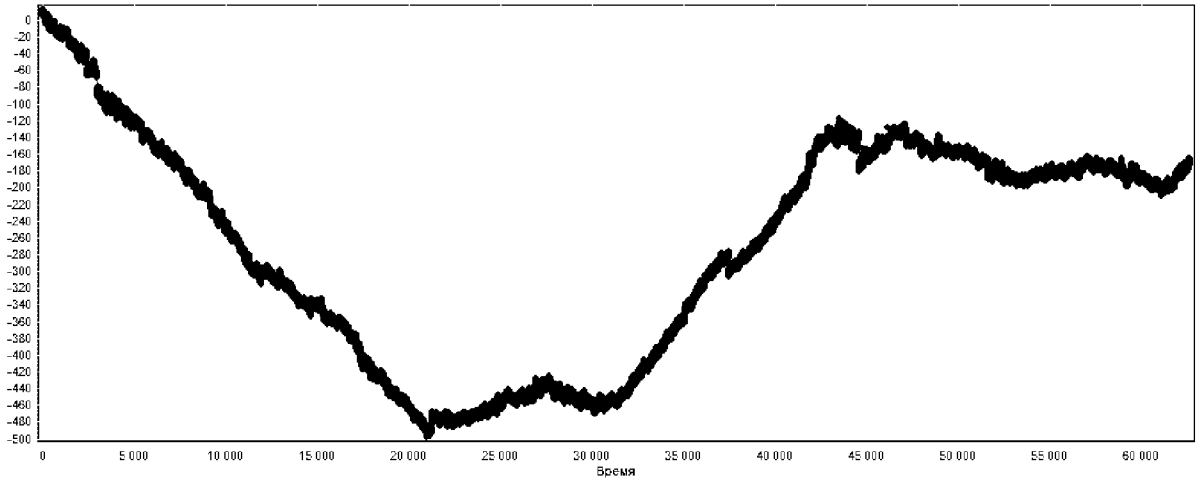



Рис. 4. График зависимости мгновенной оценки расхождения ШВ от времени (по оси ординат расхождение указано в нс)

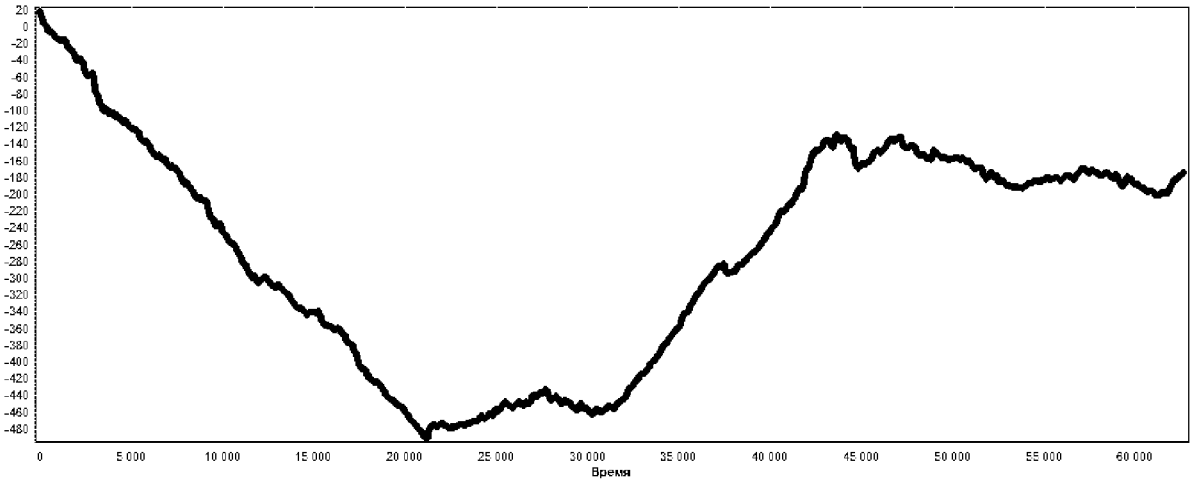



Рис. 5. График зависимости сглаженной оценки расхождения ШВ от времени (по оси ординат расхождение указано в нс)

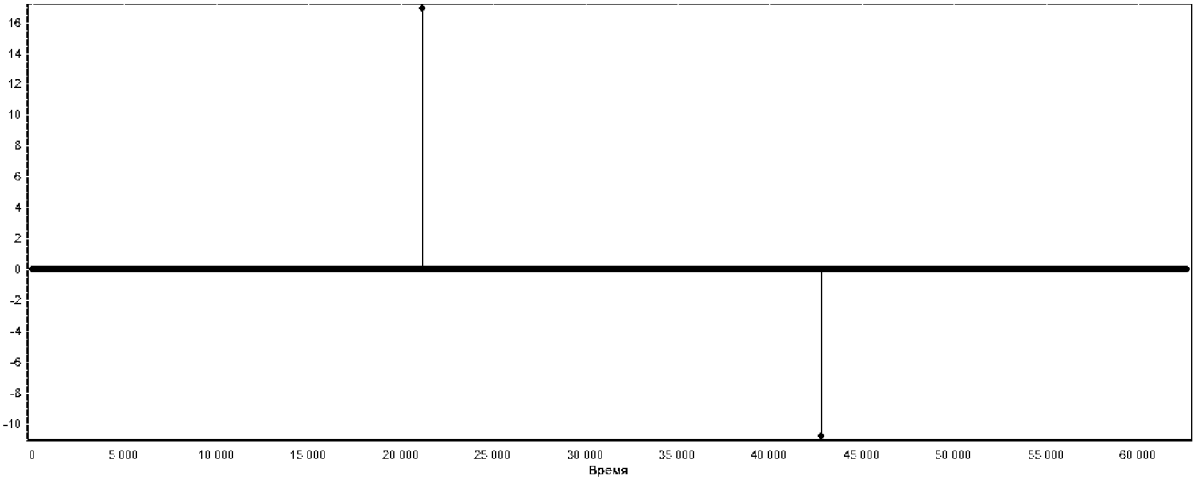



Рис. 6. График зависимости кода синтезатора частоты на выходе схемы управления от времени

## Заключение

В работе предложена модель системы мониторинга сигналов средств синхронизации системы единого времени и приведен алгоритм управления частотой опорного генератора системы. Приведенный алгоритм был реализован в программе, с помощью которой был проведен эксперимент. На основе полученных результатов можно сделать вывод о возможности построения измерительной системы высокой точности с применением высокостабильного опорного генератора.

## Литература

1. Акулов И.И., Кузнецов В.П., Найдеров В.З., Притычин А.Г., Хомяков Э.Н., Шур Л.М. Основы теории единого измерения времени. М.: МО СССР, 1975. 384 с.
2. Тихонов В.И. Статистическая радиотехника. 2–е изд. М.: Радио и связь. 1982. 624 с.
3. Бродин В.Б., Калинин А.В. Системы на микроконтроллерах и БИС программируемой логики. М.: Издательство ЭКОМ, 2002. 400 с.

# СПОСОБ ФОРМИРОВАНИЯ СИГНАЛОВ СПУТНИКОВОЙ РАДИОНАВИГАЦИОННОЙ СИСТЕМЫ ГЛОНАСС

**А.А. Скобелин, А.С. Бандура**
**Научный руководитель – д.т.н., профессор В.Л. Ткалич**

В статье рассматривается способ формирования навигационного сигнала космической навигационной системы ГЛОНАСС, который может быть использован для построения имитатора сигналов спутниковой радионавигационной системы в целях калибровки приемников космических навигационных сигналов.

## Введение

Необходимость точных измерений в территориально разнесенных точках требует формирования в точке измерения высокостабильной шкалы времени, привязанной к шкале времени UTC(SU) с высокой точностью. Подобные задачи решает аппаратура синхронизации. В настоящее время все большее распространение получает аппаратура синхронизации, работающая на основе навигационных сигналов спутниковых радионавигационных систем (СРНС) ГЛОНАСС и GPS.

Погрешность привязки шкалы времени потребителя по сигналам СРНС зависит от многих факторов, основными из которых являются:

- погрешность синхронизации бортовой шкалы космического аппарата (КА) с Государственным эталоном времени и частоты;
- погрешность измерения, вносимая задержками распространения сигнала в ионосфере;
- погрешность, вносимая алгоритмами обработки навигационных данных.

Первые два вида погрешности могут быть устранены техническими методами (например, излучением на двух частотах для компенсации ионосферных задержек). Для устранения третьего вида погрешности необходимо иметь навигационное поле с заранее известными характеристиками (и с возможностью их оперативного изменения), чтобы точно определять погрешность, вносимую математической обработкой навигационной информации, и оценить эффективность примененного алгоритма.

В настоящее время для построения приемников СРНС используют, в основном, математический аппарат, основанный на стандартных моделях атмосферы, Земли и т.д. В данной работе предлагается метод формирования навигационного радиосигнала СРНС, на основе которого возможно будет построить имитатор сигналов СРНС.

## Краткие сведения о навигационных радиосигналах системы ГЛОНАСС

Каждый спутник системы ГЛОНАСС излучает фазоманипулированные навигационные радиосигналы в диапазоне L1 (1602 МГц) и L2 (1246 МГц).

В радиолинии частотного диапазона L1 спутники системы ГЛОНАСС излучают навигационные радиосигналы двух типов: стандартной и высокой точности (СТ- и ВТ-сигнал, соответственно). Сигнал стандартной точности предназначен для использования гражданскими потребителями, и предоставляемое им обслуживание доступно всем владельцам аппаратуры потребителей ГЛОНАСС. Сигнал высокой точности модулирован специальным кодом и не рекомендован к использованию без согласования с МО РФ.

В радиолинии диапазона L2 в настоящее время передается только ВТ-сигнал, поэтому гражданские потребители не могут использовать метод двухчастотной компенсации ионосферной погрешности.

В системе ГЛОНАСС номинальное значение рабочих частот радиосигналов навигационных спутников (НС): $f_{Li}^{k} = f_{0i} + k\Delta f_i$, где $i$ = 1, 2 – номер диапазона частот; $k$ =

0,24 – номер частотного канала (литер); $f_{01}$ = 1602 МГц; $f_{02}$ = 1246 МГц; $\Delta f_{01}$ = 562,5 кГц; $\Delta f_{02}$ = 437,5 кГц. Канал $k$ = 0 не предназначен для использования потребителями системы ГЛОНАСС. Он применяется наземной подсистемой управления для проверки резервных спутников на орбите при восполнении орбитальной группировки. Сведения о распределении частотных каналов $k$ = 1,24 между спутниками, расположенными в орбитальных рабочих точках с номерами $m$ = 1,24, содержатся в альманахе системы.

Навигационный радиосигнал, передаваемый каждым НС, представляет собой многокомпонентный фазоманипулированный сигнал. Для получения высокой точности измерений задержки распространения сигнала излучаемый сигнал модулируется дальномерным кодом стандартной точности (СТ-код), представляющим периодическую последовательность максимальной длины. Для диапазона L1 тактовая частота формирования дальномерного кода $f_{ст}$ = 511 кГц, период повторения $T_{п.к.}$ = 1 мс.

Для передачи навигационной (служебной) информации используется модуляция двоичной последовательностью (кодом служебной информации (СИ-код)) с тактовой частотой $f_{СИ}$ = 50 Гц. СИ-код представляет собой преобразованную цифровую последовательность навигационных данных, передаваемых аппаратурой НС потребителям системы ГЛОНАСС. Дальномерный СТ-код представляет собой М-последовательность. Это сравнительно короткая псевдослучайная последовательность (ПСП) (длина $L$ = 511 элементов) обеспечивает быстрый поиск дальномерного сигнала и приемлемую точность измерения дальности до НС с соответствующей неоднозначностью.

Код метки времени представляет собой укороченную ПСП. Длина ПСП МВ равна тридцати символам с длительностью 10 мс каждый. Эта метка позволяет осуществлять строчную синхронизацию, а также устранять неоднозначность дальномерных измерений [1].

### Способ формирования навигационного радиосигнала

В основе предлагаемого способа лежит применение схем фазовой автоподстройки частоты (ФАПЧ) и прескайлеров (делителей частоты высокочастотных сигналов) для формирования высокочастотного навигационного сигнала. Блок-схема формирователя навигационных сигналов, модулированных ВТ-кодом, в диапазоне L2 представлена на рис. 1. Приведенная схема может быть использована также и для формирования сигналов системы ГЛОНАСС, модулированных СТ-кодом (при изменении коэффициентов деления прескайлеров).

Входными сигналами схемы являются: импульсы с частотой следования 1 Гц, полученные делением эталонной частоты 5 МГц, синусоидальный сигнал частотой 5 МГц.

Источником сигнала высокой частоты, используемого для формирования несущей, служит генератор, управляемый напряжением (ГУН). ГУН включен в кольцо схемы ФАПЧ [2], имеющей входной прескайлер, что позволяет, во-первых, использовать в качестве опорной частоты сигнал с относительно низкой частотой (0,4375 МГц), и, во–вторых, оперативно изменять литерные частоты путем изменения коэффициента деления входного прескайлера сигнала ГУН. Модуляция сигнала ВТ-кодом производится на частоте 140 МГц, которая формируется методом прямого умножения эталонной частоты 5 МГц. Это позволяет избежать конструктивных проблем, возникающих при работе с сигналами СВЧ-диапазона. Модулированный сигнал смешивается с ВЧ-сигналом, и из смеси выделяется сумма частот с заданной полосой. Таким образом, все манипуляции производятся над сигналом частотой 140 МГц, что позволяет избежать внесения дополнительных шумов в результирующий сигнал.

Интерес представляет также схема синтезатора тактовой частоты ПСП 5,11 МГц, которая приведена на рис. 2. К этому сигналу предъявляются требования высокой стабильности, поэтому он формируется также из эталонной частоты. Спектр смеси импульсов, поступающей на вход частотно-избирательного фильтра, построенного на основе кварцевого фильтра, имеет в своем составе гармонику с частотой 10,22 МГц, которая выделяется фильтром и делится на 2. Так как операцию умножения можно выполнить на цифровых элементах (операция исключающего ИЛИ), потери амплитуды сигнала, вызываемые аналоговыми фильтрами схем умножения, отсутствуют.

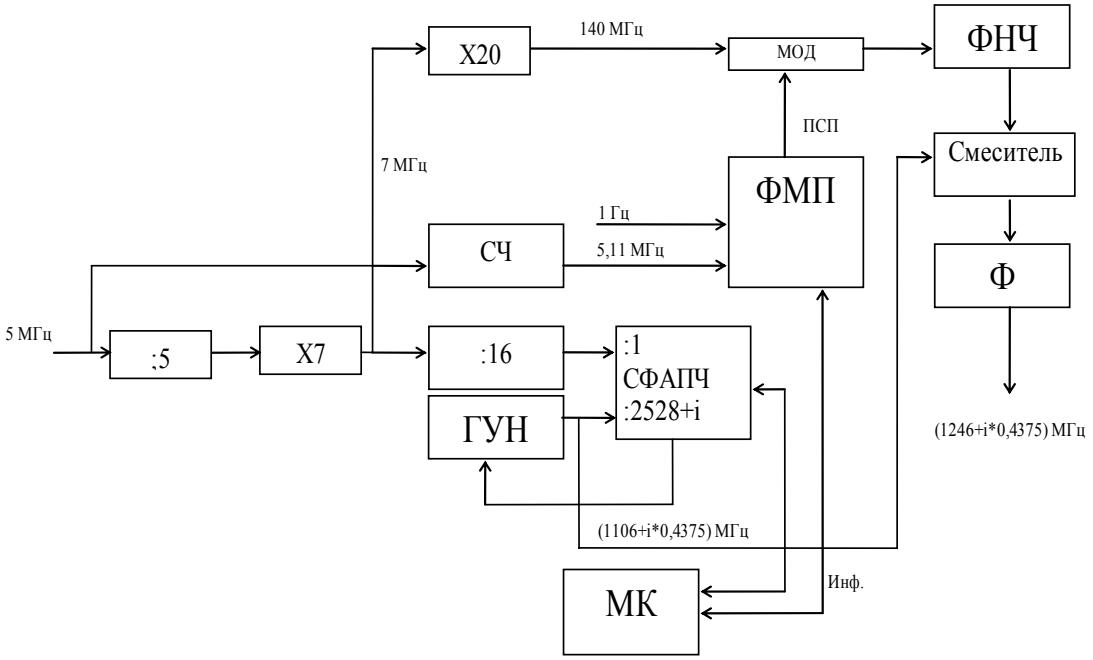



Рис. 1. Блок–схема формирователя навигационных радиосигналов СРНС ГЛОНАСС; СЧ – синтезатор частоты; ГУН – генератор управляемый напряжением; ФНЧ – фильтр низких частот (5,11 МГц); СФАПЧ – схема фазовой автоподстройки частоты; ФМП – формирователь модулирующей последовательности; МОД – модулятор; МК – микроконтроллер; Ф – полосовой фильтр выходного сигнала; Инф. – навигационная информация

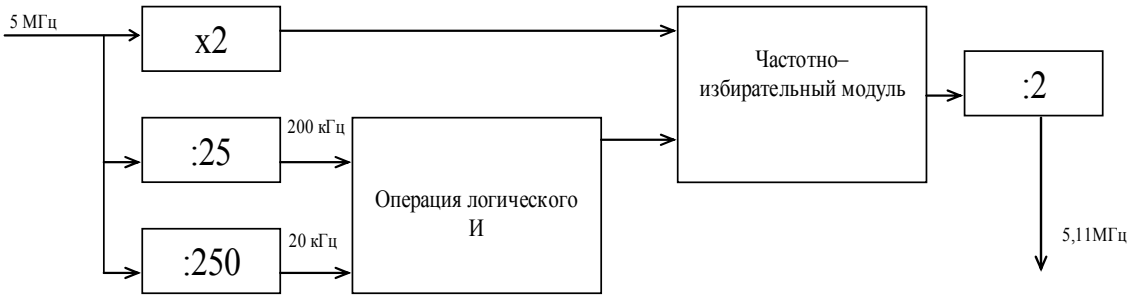



Рис. 2. Блок–схема синтезатора частоты 5,11 МГц

## Анализ схемы

Так как все сигналы в предлагаемой схеме формируются методами прямого умножения и деления эталонной частоты на основе цифровой логики, она обладает низким уровнем фазовых шумов. Уровень фазовых шумов выходного сигнала, по большей части, определяется уровнем шумов сигнала выбранного ГУН и шумами, вносимыми схемой ФАПЧ. Чем меньше крутизна перестройки ГУН (коэффициент связи между напряжением управления и изменением частоты), тем точнее установка частоты и меньше возможная отстройка от номинала. Так как допускаемая погрешность фазовой манипуляции навигационного сигнала составляет 0,2 радиан (относительная погрешность – 0,063), то погрешностями, вносимыми модулятором, можно пренебречь.

Кроме того, управляя фазой выходного сигнала, становится возможным вносить задержки сигналов друг относительно друга и, тем самым, проверять эффективность решения задачи ориентации потребителя в пространстве.

## Заключение

В работе приведена структурная схема и описаны принципы формирования радионавигационных сигналов системы ГЛОНАСС. Проведен краткий анализ схемы, который позволяет полагать, что на ее основе можно построить имитатор навигационных сигналов СРНС, с помощью которого будет возможно разрабатывать алгоритмы обработки навигационных данных и оценивать их эффективность.

## Литература

1. Глобальная спутниковая навигационная система ГЛОНАСС. / Под ред. Харисова В.Н., Перова А.И., Болдина В.А. М.: ИПРЖР, 1999.
2. Шахгильдян В.В., Ляховкин А.А., Карякин В.Л. и др. Системы фазовой синхронизации с элементами дискретизации. / Под ред. В.В. Шахгильдяна. 2-е изд., доп. и пер.. М.: Радио и связь, 1989. 320 с.

# РАСПОЗНАВАНИЕ ОБРАЗОВ В ИЗОБРАЖЕНИЯХ

**В.Г. Иванов**
**Научный руководитель – к.т.н., доцент Б.А. Крылов**

## Введение

Интерес к вопросу распознавания образов велик. Уже сегодня можно в повседневной жизни встретить плоды научных открытий: программные средства по распознаванию текстов, системы распознавания лиц, номеров проезжающих по шоссе машин, а также системы распознавания эмоционального состояния человека по выражению лица [1]. В силу сложности поставленных задач широкое применение получили узкоспециализированные системы распознавания. Такие системы делать намного проще: меньшая предметная область и проще требования к системе. Ниже предлагается решение, относящееся к такому типу проектов.

## Основная часть

Представление изображения как массива ячеек (пикселей) удобно для хранения его в файле, а также для формирования рисунка. Но это представление никак нельзя считать удобным при организации эффективной процедуры распознавания уже хотя бы потому, что низкий уровень абстракции (пиксели) приведет к неоправданной громоздкости вычислений. И, что самое важное, в процессе распознавания «на равных» будут учитываться как действительно важные части, влияющие на смысл, так и несущественное. В итоге работа с пикселями приведет к необходимости предварительно создавать специальную библиотеку, при помощи которой потом можно будет заняться делом.

У каждого символа есть свои особенности начертания. Любой алгоритм будет учитывать это. Так почему бы сразу не перейти на абстрактный уровень этих особенностей? Именно это я и предлагаю сделать.

На мой взгляд, механизм распознавания должен оперировать не точками яркости, а структурными особенностями примитивов и их сочетаний в изображении. Если же в изображении информация представлена текстом, то распознавание должно проводиться со структурными особенностями символов. Для этого мы должны, во-первых, формализовать особенности начертания примитивов, а во-вторых, расшифровать само изображение – из растра «перевести» содержимое в карту структурных связей. В таком контексте распознавание – это ни что иное, как поиск известных структур на карте связанных структур.

Что касается формализации примитивов (в том числе и символов), то здесь, опираясь на классификации по различным признакам (например, по общим частям), надо установить порядки начертания всех символов. На рис. 1 представлен один из результатов классификации.

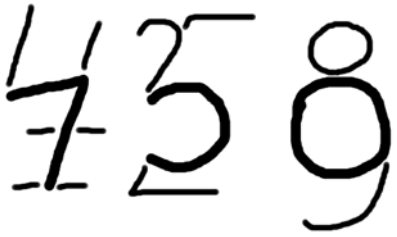


Рис. 1. Результат выявления общих частей у арабских цифр

Для расшифровки изображения в карту связанных структур самым подходящим является представление, состоящее из структуры узловых точек (с обязательным учетом координат) и связующих линий – граф. Отсекая все лишнее, мы получаем ясную и относительно простую структуру, которую можно обрабатывать дальше.

Под дальнейшей обработкой подразумеваются как простые действия со связующими линиями (например, вычисление одной усредненной линии из двух), так и сложные действия (например, наложение нескольких карт слоями) и, конечно же, поиск структур на карте. Промежуточными результатами работы должны стать предположения системы о закодированной информации, основанные на точности совпадения структуры частей карты с эталонными структурами. Предположения обобщаются и образуют варианты трактовки карты, которых в результате работы может оказаться больше одной.

Учитывая все сказанное, структуру системы распознавания можно в самом общем виде представить как группу связанных подсистем создания предположений и их принятия, что наглядно проиллюстрировано на рис. 2.


Рис. 2. Общая структура системы распознавания

Подсистема создания предположений получает первоначальную структуру карты с расшифрованным в нее изображением от синтаксического анализатора и начинает ее обработку в поисках совпадений с эталонами. В нетривиальном случае для успешного поиска появится необходимость в изменении структуры по правилам модификации. Эти правила являются результатом «опыта» работы подсистемы создания предположений и в процессе работы формируются подподсистемой добавления правил модификации. В результате всех действий подсистема должна выработать предположение. В дальнейшем это предположение обрабатывается подсистемой принятия предположений. Главная особенность этой подсистемы заключается в способности работать на нескольких логических уровнях, переходя от одного к другому (точнее, опускаясь из одного в другой) при помощи обобщения и разобщения полученных предположений.

Хочу подчеркнуть, что восприятие даже 2-х мерного изображения с ограниченным числом символов (текст, который вы сейчас читаете) требует от нашего мозга способности «переключаться» с уровня букв – в слова, а из слов – в фразы, т.е. на смысловой уровень. Соответственно, по мере повышения уровня обобщения предположений подсистема принятия предположений вырабатывает варианты трактовки структур карты.

## Заключение

Распознавание образов – довольно сложный процесс, требующий учета множества нюансов в проектируемой системе. Но, по сути, они второстепенны. А главное, как мне видится, заключается в относительно простом принципе распознавания, который можно представить как последовательность ограниченного числа действий (а также действий, вытекающих из них) над структурами, организованную на нескольких логических уровнях.

## Литература

1. http://www.rambler.ru/news/science/0/9789863.html

# ОПРЕДЕЛЕНИЕ СТАБИЛЬНОСТИ ТРАЕКТОРИИ ПРОЦЕССА В ФАЗОВОМ ПРОСТРАНСТВЕ ПРИ ПОМОЩИ РЕКУРРЕНТНОГО АНАЛИЗА

**В.Б. Киселев**
**Научный руководитель – к.т.н., доцент Б.А. Крылов**

Предложен способ количественной оценки стабильности фазовой траектории изучаемого процесса при помощи диаграмм расстояний (подвид рекуррентных диаграмм) и их количественного анализа. Приведены примеры применения способа к модельным и реальным системам, дана оценка возможностей использования.

## Введение

Исследования сложных систем, как природных, так и искусственных, показали, что в их основе лежат нелинейные процессы, тщательное изучение которых необходимо для понимания и моделирования сложных систем. В последние десятилетия набор традиционных (линейных) методик исследования был существенно расширен нелинейными методами, полученными из теории нелинейной динамики и хаоса. Однако большинство методов нелинейного анализа требуют либо достаточно длинных, либо стационарных рядов данных, которые довольно трудно получить из природы. Более того, было показано, что данные методы дают удовлетворительные результаты в основном для идеализированных моделей реальных систем [1].

Рекуррентный анализ [2–4] – достаточно молодой и динамично развивающийся подход к анализу сложных систем, не требующий длинных или стационарных временных рядов. Рекуррентные диаграммы позволяют судить о характере протекающих в системах процессов, наличии и влиянии шума, дрейфа, наличии состояний повторения и замирания (ламинарность), совершении экстремальных событий, наличии скрытой периодичности и цикличности. Количественный анализ рекуррентных диаграмм позволяет сопоставить диаграмме некоторые численные меры, основанные на плотности рекуррентных точек, диагональных и вертикальных (горизонтальных) линий. Следует отметить, что пока не создано удовлетворительной теории применения рекуррентных диаграмм и их количественных мер; этот метод сам по себе представляет собой поле для исследований.

В настоящей работе предложен способ количественной оценки стабильности фазовой траектории изучаемого процесса при помощи диаграмм расстояний (подвид рекуррентных диаграмм) и их количественного анализа. Приведены примеры применения способа к модельным и реальным системам, дана оценка возможностей использования.

## Рекуррентные диаграммы

Рекуррентные диаграммы были предложены для отображения траектории $\vec{x}_t \in \mathbf{R}^m \ (i = 1 \ldots N)$ в $m$-мерном фазовом пространстве на двумерную двоичную матрицу размером $N \times N$. Единица в ячейке матрицы соответствует повторению состояния (проход траектории через одну и ту же точку фазового пространства) при некотором времени $i$ в некоторое другое время $j$, а обе координатные оси диаграммы являются осями времени. Математически это выражается следующим образом:

$$\mathbf{R}_{i,j}^{m,\varepsilon_i} = \Theta\left(\varepsilon_i - \left\|\vec{x}_i - \vec{x}_j\right\|\right), \quad \vec{x} \in \Re^m, \quad i, j = 1 \ldots N, \tag{1}$$

где $N$ – количество рассматриваемых состояний $x_i$, $\varepsilon_i$ – размер окрестности точки $\vec{x}$ в момент $i$, $\|\cdot\|$ – норма (расстояние), $\Theta(\cdot)$ – функция Хэвисайда. Графически рекур-

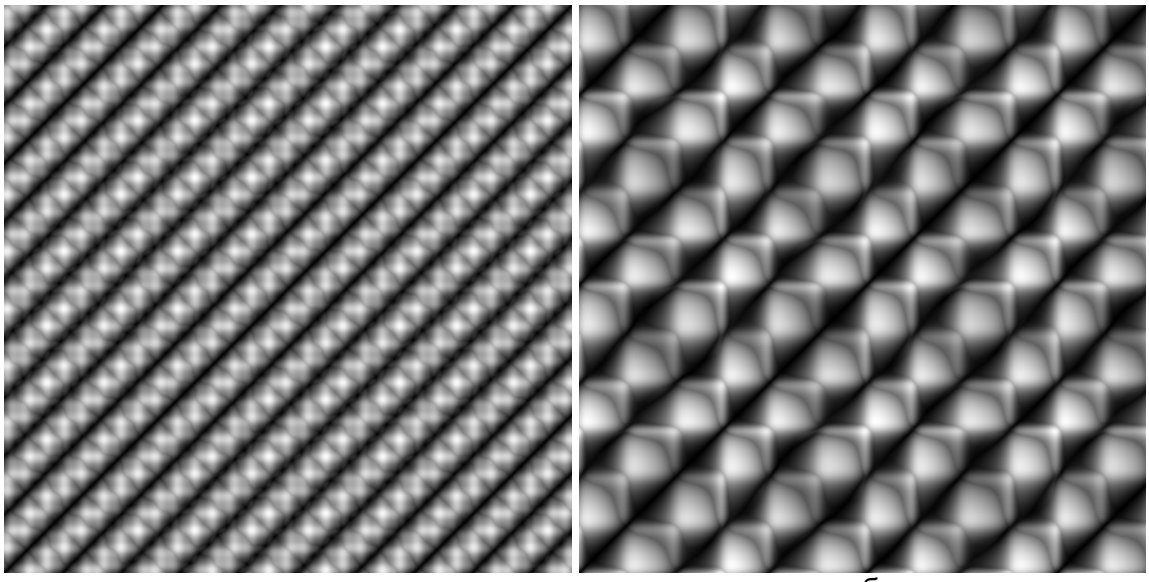
рентная диаграмма может быть представлена монохромным изображением, где единице соответствует черная точка.

При наличии только одномерного ряда $u_t$ эквивалентная траектория в $m$-мерном фазовом пространстве может быть восстановлена по методу временных задержек Такенса [5] $\vec{x}(t) = \left(u_i, u_{i+\tau}, \ldots, u_{i+(m-1)\tau}\right)$, где $m$ – размерность вложения, $\tau$ – временная задержка (реальная временная задержка определяется как $\tau \cdot \Delta t$). Топологические структуры восстановленной траектории сохраняются, если $m \le 2d+1$, где $d$ – размерность аттрактора. На практике оказывается, что в большинстве случаев аттрактор может быть восстановлен и при $m \le 2d$ [6]. Задержка $\tau$, как правило, выбираются априорно.

Размер окрестности $\varepsilon_i$ определяет радиус окрестности в фазовом пространстве с центром в точке $x_i$. Если точка $x_j$ попадает внутрь данной окрестности, то такое состояние считается подобным состоянию $x_i$, и на диаграмме устанавливается $\mathbf{R}_{i,j} = 1$. Радиус $\varepsilon_i$ может быть постоянным для всех $x_i$ либо определяться для каждой точки индивидуально, чтобы в получаемую окрестность всегда попадало определенное количество подобных состояний. В этой работе используется постоянное значение $\varepsilon_i$, что приводит к получению симметричной рекуррентной диаграммы относительно линии $\mathbf{R}_{i,j} = 1 \quad (i = j)$. Для вычисления расстояний между точками фазового пространства используется максимальная норма $L_\infty \equiv \sup|v_k|$.

Рисунок рекуррентной диаграммы отображает поведение процесса во времени и позволяет сделать выводы о его характере по ее топологии и текстуре.

Збилут (Zbilut) и Вебер (Webber) разработали количественный анализ рекуррентных диаграмм (recurrence quantification analysis, RQA) [7] для определения численных показателей рекуррентной диаграммы. Они предложили меры, использующие плотность рекуррентных точек и диагональные структуры диаграммы: показатель подобия (*RR*), детерминизм (*DET*), максимальная длина диагональных линий (*L*), энтропия (*ENTR*), тренд (*TREND*).



а б

Рис. 1. Фрагменты диаграмм расстояний: а – рассматриваемая в работе модельная система, б – две компоненты уравнения Ван-дер-Поля для стандартных параметров

Если исключить проверку попадания точек в $\varepsilon$-окрестность других точек, получим диаграмму расстояний, каждая точка которой несет информацию о расстоянии между точками $x_i$ и $x_j$ траектории [8, 9]:

$$\mathbf{D}^m_{i,j} = \left\| \vec{x}_i - \vec{x}_j \right\|. \qquad (2)$$

Такая диаграмма удобна тем, что позволяет проводить исследования для различных значений $\varepsilon$. Графически она может быть изображена при помощи отображения на некоторую цветовую палитру, например уровней серого (рис. 1). Также диаграммы расстояния могут быть интересны с эстетической точки зрения, так как сложнопериодические процессы позволяют получить довольно интересные изображения.

### Количественный анализ диаграмм. Показатель подобия

Существует несколько общеизвестных мер количественного анализа рекуррентных диаграмм, основанных на подсчете плотности точек, диагональных и вертикальных (горизонтальных) линий, позволяющие численно выразить структуры на рекуррентной диаграмме. Удовлетворительная теория использования этих мер пока не разработана. Одна из этих мер – показатель подобия (recurrence rate) [7] – представляет собой меру плотности рекуррентных точек на диаграмме:

$$RR = \frac{1}{N^2} \sum_{i,j=1}^{N} \mathbf{R}^{m,\varepsilon}_{i,j}. \qquad (3)$$

Очевидно, что для одной и той же траектории показатель подобия является функцией от радиуса окрестности $\varepsilon$:

$$RR(\varepsilon) = RR(\varepsilon_k), \quad \begin{array}{c} \varepsilon_k = \varepsilon_0 + k \cdot \Delta\varepsilon, \quad \varepsilon_k < \max(\mathbf{D}_{i,j}), \\ k = 0,1,\ldots, \qquad i,j = 0 \ldots N-1, \end{array} \qquad (4)$$

где $\varepsilon_0$ – начальное значение радиуса окрестности, $\Delta\varepsilon$ – шаг увеличения, $\mathbf{D}_{i,j}$ – диаграмма расстояний (2) размером $N \times N$. Значение $\varepsilon_k$ изменяется с шагом $\Delta\varepsilon$ в пределах от $\varepsilon_0$ до максимального расстояния между точками траектории в фазовом пространстве. Скорость изменения $RR(\varepsilon)$ равна

$$RR'(\varepsilon) = RR(\varepsilon_k) - RR(\varepsilon_{k-1}), \quad k = 1 \ldots K-1, \qquad (5)$$

где $K$ – количество значений $\varepsilon_k$, использованное для вычисления $RR$. Очевидно, что при $RR'_{0 \ldots K-2} \approx RR' \approx \text{const}$ происходит строго линейное нарастание $RR(\varepsilon)$, что говорит о равномерном распределении точек временного ряда по траектории ($RR(\varepsilon)$ при каждом приращении $\varepsilon_k$ увеличивается на $RR' = \text{const}$).

Строго говоря, такая ситуация возможна, во-первых, если процесс обладает гладкой, без флуктуаций траекторией, сходящейся к точке, и, во-вторых, если задержка временного ряда $\Delta t \to 0$.

Периодические процессы, встречающиеся на практике, порождают траектории различной степени сложности, зачастую отягощенные шумами, случайными флуктуациями и разного рода искажениями. Если принять, что траектория гладкая, задержка временного ряда $\Delta t \to 0$, то кривая роста $RR(\varepsilon)$ будет представлять собой несколько сопряженных прямых отрезков, количество которых будет зависеть от формы траектории. Шумы и случайные флуктуации системы приводят к увеличению скорости возрастания $RR(\varepsilon)$ за счет появления на рекуррентной диаграмме отдельно стоящих рекуррентных точек, вертикальных и горизонтальных линий так, что контуры рисунка диаграммы размываются.

Таким образом, по изменению размаха скорости

$$\Delta RR' = \max(RR') - \min(RR') \equiv STAB \qquad (6)$$

можно судить о стабильности исследуемой траектории.

## Моделирование

Проследим динамику изменения $RR(\varepsilon)$ и $RR'(\varepsilon)$ для траектории системы

$$\begin{cases} x = \cos(t)\cdot r \\ y = \sin(t)\cdot r \end{cases}, \quad r = \mu\cdot\sin(t\cdot\pi + \eta\cdot\mu\cdot \mathrm{RND}) + \eta\cdot\mu\cdot\mathrm{RND}, \qquad (7)$$

где $\eta$ – уровень равномерно распределенного шума $\mathrm{RND}$, вносящего нестабильность в траекторию, $\mu = 0.2$ – коэффициент толщины окружности.

Фазовые портреты системы (7) для значений $\eta = 0.0$, $\eta = 0.2$ и $\eta = 0.7$ представлены на рис. 2. Временной ряд сгенерирован с временной задержкой $\Delta t = 0.061$ длиной порядка 3000 точек.

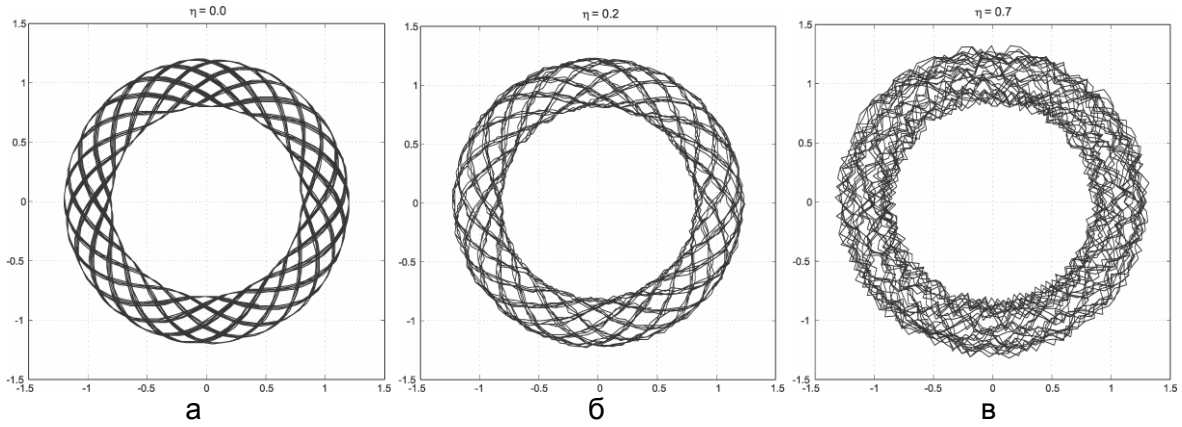
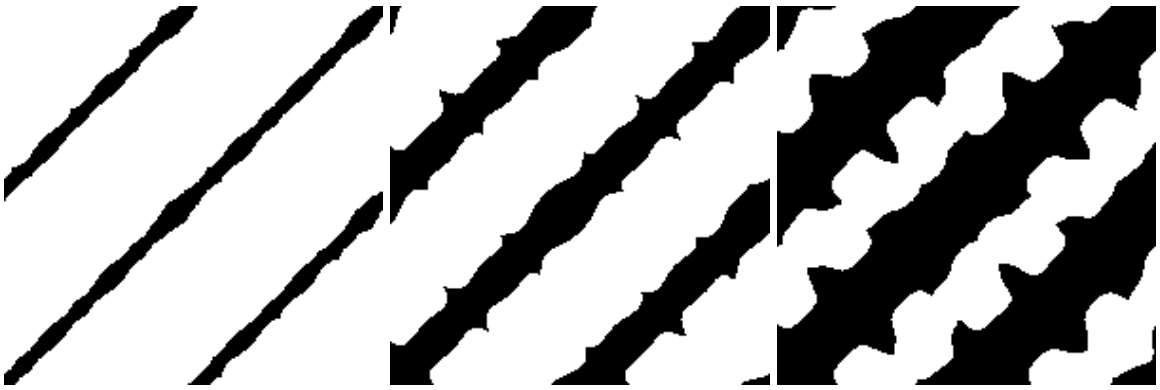



Рис. 2. Фазовые портреты: а – $\eta = 0.0$, б – $\eta = 0.2$, в – $\eta = 0.7$

Система (7) порождает довольно сложную периодическую траекторию; видны искажения (рис. 2б, в), вносимые равномерно распределенным шумом. На рис. 1а показан фрагмент диаграммы расстояний для системы (7) при $\eta = 0.0$.

Рассмотрим фрагменты рекуррентных диаграмм для системы (7) при $\eta = 0.0$ (рис. 3) и $\eta = 0.7$ (рис. 4) для разных $\varepsilon$. Из диаграмм видно, что плотность рекуррентных точек в случае зашумленной системы растет сильнее за счет появления отдельно стоящих рекуррентных точек и вертикальных и горизонтальных линий; края относительно гладкой структуры (рис. 3) становятся «мохнатыми» (рис. 4), размываются. Визуально заметно появление отдельных точек и линий, отображающих шумовую составляющую.

Рассмотрим графики изменения $RR(\varepsilon)$ для системы (7) при значениях $\eta = 0.0, 0.2, 0.7$. Вычисление $RR(\varepsilon)$ проводилось, начиная со значения $\varepsilon_0 = 0.01$, с шагом $\Delta\varepsilon = 0.05$. На рис. 5 представлены графики $RR(\varepsilon)$ и $RR'(\varepsilon)$ для системы (7) при значениях $\eta = 0.0, 0.2, 0.7$. Видно, что график $RR(\varepsilon)$ разбит на две части – практически линейный рост в диапазоне $\varepsilon \approx 0.2\ldots1$, затем – переход к более крутому росту после $\varepsilon = 1.1$. Такая форма графика объяснима формой фазовой траектории системы (рис. 2а) – в районе значения $\varepsilon \approx 1$ в окрестность $i$-й точки траектории начинают попадать точки из другой половины окружности, что и обуславливает более интенсивный рост значений $RR$.

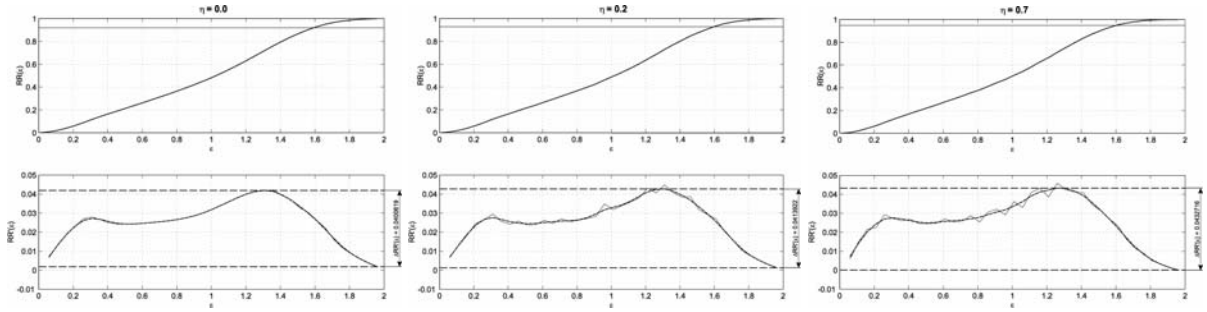


а б в

Рис. 3. Рекуррентные диаграммы модельной системы при $\eta = 0.0$:
а – $\varepsilon = 0.3$, б – $\varepsilon = 0.7$, в – $\varepsilon = 1.1$

а б в

Рис. 4. Рекуррентные диаграммы модельной системы при $\eta = 0.7$:
а – $\varepsilon = 0.3$, б – $\varepsilon = 0.7$, в – $\varepsilon = 1.1$




а б в

Рис. 5. Графики изменения $RR$ (сверху) и $RR'$ (снизу) для модельной системы при ($\varepsilon_0 = 0.01$, $\Delta\varepsilon = 0.05$): а – $\eta = 0.0$, б – $\eta = 0.2$, в – $\eta = 0.7$

Резкие выпады на графиках $RR'(\varepsilon)$ (рис. 5б, в) обусловлены несовершенством модели и дискретностью временного ряда и ряда $\varepsilon_k$. Для нивелирования выпадов целесообразно проводить аппроксимацию значений $RR'$. На приведенном графике аппроксимированная кривая изображена толстой пунктирной линией; для ее получения использовалась функция аппроксимации сплайнами, входящая в стандартную библиотеку системы Matlab.

Следует отметить влияние шума на вид графиков $RR(\varepsilon)$, выражающееся в более крутом росте значений $RR(\varepsilon)$, особенно во второй половине графиков. Для наглядно-

сти на всех трех графиках проведена линия пересечения кривой с прямой, соответствующей $\varepsilon_k = 1.6$. Видно, что при $\eta = 0.7$ эта линия расположена выше.

Значения размаха (6) $\Delta RR' \equiv STAB$ составили:

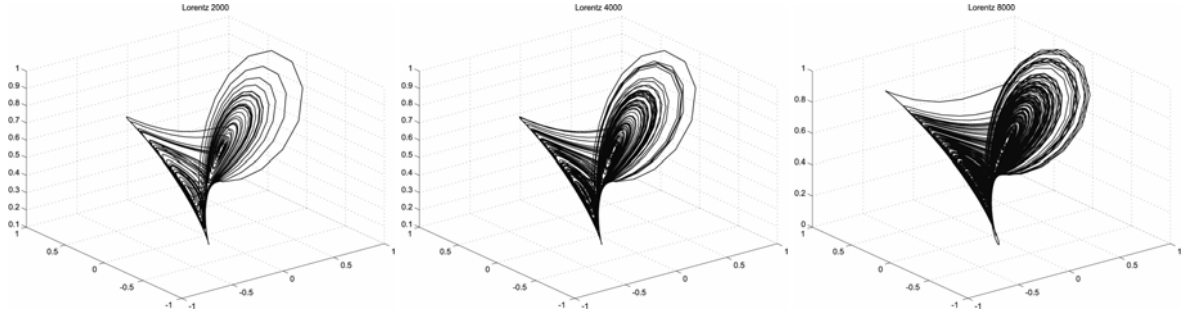
$STAB_{\eta=0.0} = 0.0400619$;

$STAB_{\eta=0.2} = 0.0413922$;

$STAB_{\eta=0.7} = 0.0432716$.

Очевиден рост размаха с ростом нестабильности траектории. Таким образом, меньшие значения $STAB$ будут соответствовать большей стабильности траектории.

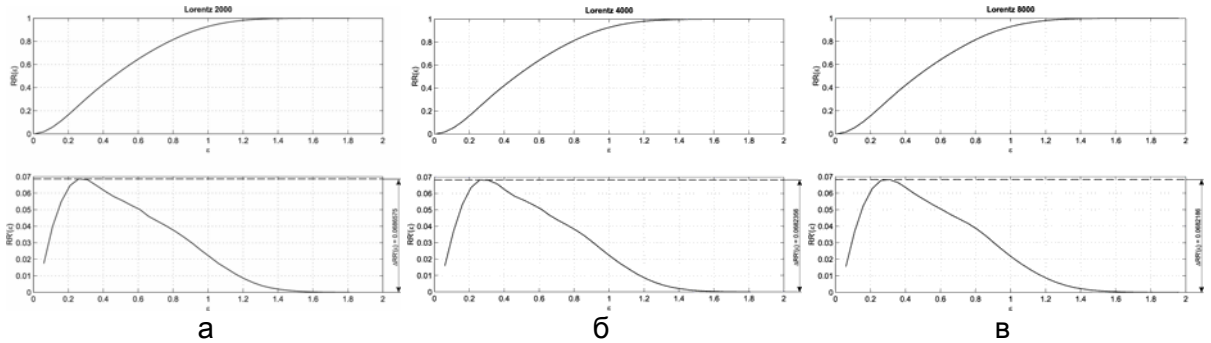
Проверим зависимость $STAB$ от длины временного ряда. Для этого используем временные ряды системы Лоренца (рис. 6).




а б в

Рис. 6. Фазовые портреты системы Лоренца (стандартные параметры $r = 28$, $\sigma = 10$, $b = 8/3$) для временных рядов длиной: а – 2000, б – 4000, в – 8000 точек

Известно, что увеличение длины временного ряда улучшает достоверность результатов. Однако может оказаться, что мера $STAB$ обладает зависимостью от длины временного ряда. На рис. 7 представлены графики $RR(\varepsilon)$ и $RR'(\varepsilon)$ для рассматриваемых рядов системы Лоренца.




а б в

Рис. 7. Графики изменения $RR$ (сверху) и $RR'$ (снизу) временных рядов системы Лоренца длиной: а – 2000, б – 4000, в – 8000 точек

Значения размаха (6) $\Delta RR' \equiv STAB$ составили:

$STAB_{N=2000} = 0.0686575$;

$STAB_{N=4000} = 0.0682356$;

$STAB_{N=8000} = 0.0682186$.

Видно, что значения $STAB$ практически равны; наблюдается лишь незначительное уменьшение по мере увеличения длины ряда. Это объясняется, в частности, хаотическим характером системы Лоренца. Также можно говорить о некотором улучшении достоверности получаемого значения. В целом можно сделать вывод о независимости

$STAB$ от длины временного ряда, что подтверждается также анализом других временных рядов.

### Различия между *STAB* и *DET*

Мера детерминизма (*DET*) [7] учитывает диагональные линии рекуррентной диаграммы с использованием частотного распределения длин диагональных линий $P^{\varepsilon}(l) = \{l_i; i = 1 \ldots N_l\}$, где $N_l$ – абсолютное количество диагональных линий (каждая линия считается только один раз). Процессы со стохастическим поведением могут порождать очень короткие диагонали либо вообще не порождать их, в то время как детерминистские процессы дают длинные диагонали и малое количество отдельных рекуррентных точек. Таким образом, отношение рекуррентных точек, составляющих диагональные структуры, к их общему количеству $DET = \dfrac{\sum_{l=l_{\min}}^{N} lP^{\varepsilon}(l)}{\sum_{i,j}^{N} \mathbf{R}_{i,j}^{m,\varepsilon}}$, называется мерой детерминизма (determinism, *DET*) или предсказуемости системы. Следует отметить, что эта мера не имеет значения реального детерминизма процесса. Пороговое значение минимальной длины $l_{\min}$ исключает диагональные линии, образованные тангенциальным движением траектории в фазовом пространстве. Очевидно, что $l_{\min} = 1 \Rightarrow DET = 1$.

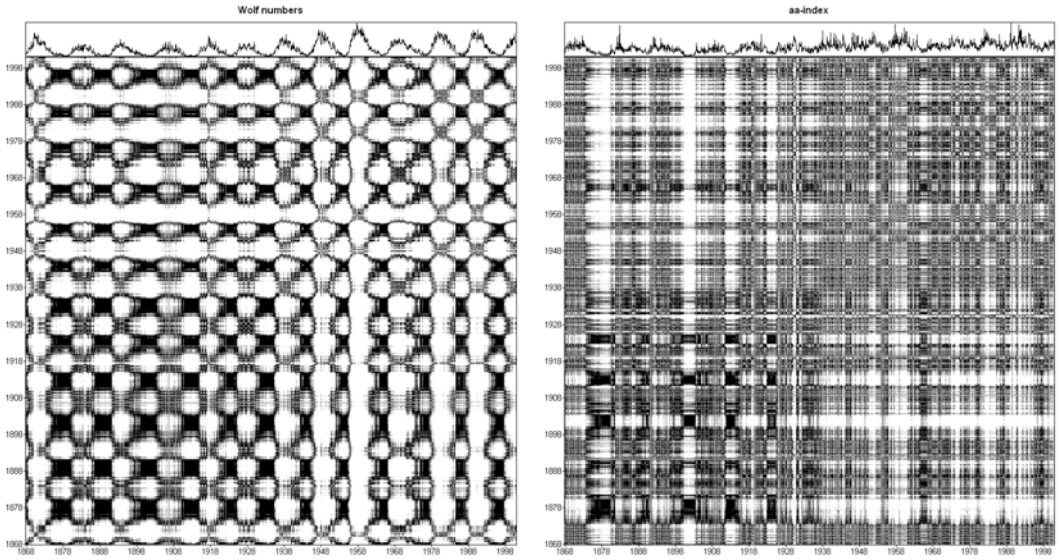
Отличие детерминизма от предлагаемой меры $STAB$ заключается в том, что первая оценивает периодичность траектории в смысле близкого прохождения участков траектории в фазовом пространстве и зависит, прежде всего, от характера изучаемой системы. Так как пороговое значение $l_{\min}$ позволяет отсечь шумовую составляющую диаграммы – т.е. отдельно стоящие точки и короткие линии – то при достаточно малом уровне шума значение $DET$ будет постоянным при одном и том же $\varepsilon$; мало того, в случае существенного увеличения зашумленности значение $DET$ будет повышаться при сохранении $\varepsilon$ за счет высокой плотности (вплоть до объединения в достаточно длинные диагональные линии) порождаемых шумовой составляющей рекуррентных точек. В то же время $STAB$ является мерой стабильности траектории и позволяет оценить влияние шумов и случайных флуктуаций вне зависимости от формы траектории (т.е. характера изучаемой системы).

### Применение *STAB*

Особенность предложенной меры заключается в том, что она может быть использована только для сравнения рядов, полученных при разных измерениях одних и тех же переменных состояния одной и той же системы или одинаковых систем. Обусловливается это тем, что другой набор переменных состояния приведет к совершенно иному фазовому портрету, как следствие – совершенно иной рекуррентной диаграмме и поведению $RR(\varepsilon)$. Данная мера может быть использована:

- для оценки стабильности системы с целью, например, подстройки ее параметров (полученное значение сравнивается с эталонным);
- для оценки влияния шумов и искажений в результате некоторых преобразований;
- для оценки стабильности связанных систем, одна из которых некоторым образом определяет поведение другой.

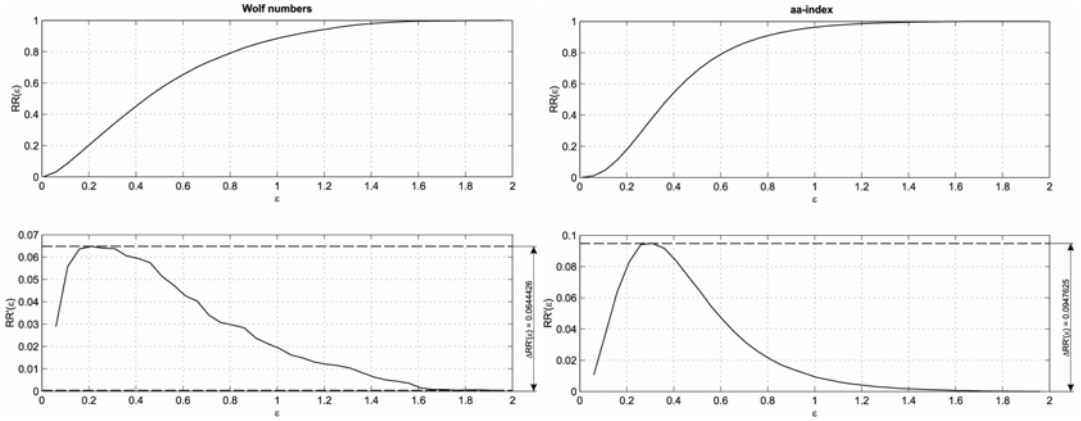
Рассмотрим пример. Солнечная активность, количественно выражаемая числами Вольфа, оказывает существенное влияние на геомагнитное поле Земли. Рекуррентная диаграмма временного ряда чисел Вольфа за период с 01.1868 г. по 12.2000 г. представлена на рис. 8а.




а б

Рис. 8. Рекуррентные диаграммы чисел Вольфа (а) и аа-индекса (б). Временные ряды взяты за период с 01.1868 по 12.2000 с шагом в 1 месяц

Для характеристики общепланетарной возмущенности геомагнитного поля Земли сконструированы различные индексы, характеризующие возмущенность поля на различных широтах. Одним из таких индексов является аа-индекс, получаемый усреднением локальных индексов, полученных в геомагнитных обсерваториях на средних широтах. Рекуррентная диаграмма ряда аа-индекса за период с 01.1868 г. по 12.2000 г. представлена на рис. 8б.




а б

Рис. 9. Графики изменения $RR$ (сверху) и $RR'$ (снизу) для чисел Вольфа (а) и аа-индекса (б) для диаграмм временных рядов с 01.1868 по 12.2000

Известно, что солнечная активность оказывает влияние на геомагнитное поле Земли. Этот факт отчетливо виден из построенных для одного временного промежутка диаграмм на рис. 8. 11-летний цикл солнечной активности явно просматривается в динамике аа-индекса. Также на обеих диаграммах видна область относительно низкой солнечной активности, завершающаяся в районе 1937 года (граница фазового перехода). Так как оба временных ряда, очевидно, отображают связанные процессы, один из которых (солнечная активность) модулирует другой (геомагнитная активность), вычисление $STAB$ для этих рядов позволит количественно оценить, насколько сильно влияние солнечной активности на геомагнитное поле на данном промежутке времени

(рис. 9). Из изложенного выше понятно, что уровень влияния будет определяться близостью $STAB_{aa}$ к $STAB_W$.

Рассмотрим полученные значения $STAB$ для рядов чисел Вольфа ($STAB_W$) и аа-индекса ($STAB_{aa}$) на полном диапазоне дат и на промежутках до 12.1936 года и после 01.1937 года.

| **Даты** | **01.1868 – 12.2000** | **01.1868 – 12.1936** | **01.1937 – 12.2000** |
|---|---|---|---|
| $STAB_W$ | 0.0644426 | 0.0534079 | 0.0308646 |
| $STAB_{aa}$ | 0.0947625 | 0.0573547 | 0.0527186 |
| $H_W$ | 0.93 | 0.873 | 0.909 |
| $H_{aa}$ | 0.92 | 0.908 | 0.748 |

Таблица. Значения $STAB$ и показателя Харста, полученные для чисел Вольфа и аа-индекса

Близкие значения $STAB$ на промежутке 01.1868–12.1936 свидетельствуют о большей связи между солнечной и геомагнитной активностями (что также видно и по рекуррентным диаграммам – см. рис. 8). В то же время с 1937 года, несмотря на возросшую солнечную активность и возросшую стабильность ряда чисел Вольфа, влияние последних на магнитосферу Земли меньше.

Полученные результаты хорошо согласуются с приведенными в [10] расчетами показателя Харста (см. в таблице значения $H_W$ и $H_{aa}$) для тех же интервалов. По-видимому, это связано со структурой потока солнечной плазмы. При умеренной и слабой солнечной активности числа Вольфа определяют структуру потока солнечной плазмы, а взаимодействие с магнитосферой Земли почти линейно. В периоды высокой солнечной активности числа Вольфа уже не определяют структуру и параметры потоков солнечной плазмы. Так, по данным авторов работы [11] и цитируемой в ней литературы, скорость солнечного ветра не меняется в фазе с числами Вольфа, а в отдельных циклах их изменения существенно различны. Это подтверждается опубликованными на сайте «Энциклопедия Кругосвет» [12] данными измерения при помощи космических аппаратов IMP-8 и Voyager-2 среднего (за 300 дней) динамического давления солнечного ветра в районе орбиты Земли (на 1 АЕ) в течение одного 11-летнего солнечного цикла с 1978 по 1991 гг.

## Заключение

В статье предложена новая мера количественного анализа рекуррентных диаграмм – мера стабильности траектории $STAB$. Достоинством данной меры является отсутствие необходимости подбора радиуса окрестности $\varepsilon$. Рассмотрены выражения для вычисления этой меры. Приведены примеры получения $STAB$ при исследовании модельных систем, показано влияние нестабильности фазовой траектории на значение меры. Дана оценка влияния длины временного ряда на значение $STAB$. Рассмотрены различия между стандартной мерой детерминизма $DET$ и $STAB$.

Рассмотрены некоторые из возможных областей применения данной меры, приведены требования к временным рядам.

## Литература


1. Manuca R., Savit R. Stationarity and nonstationarity in time series analysis. // Physica D 99 (2–3). 1996. P. 134–161.
2. Eckmann J.-P., Kamphorst S.O., Ruelle D. Recurrence Plots of Dynamical Systems. // Europhysics Letters 5. 1987. P. 973–977.
3. Киселев В.Б. Некоторые методы нелинейного анализа // Научно-технический вестник СПбГУ ИТМО. 2005. В. 20. С. 172–180.
4. Киселев В.Б. Рекуррентный анализ – теория и практика // Научно-технический вестник СПбГУ ИТМО. 2006. В. 29. С. 118–127.
5. Takens F. Detecting Strange Attractors in Turbulence. Berlin% Springer, 1981. P. 366–381.
6. Киселев Б.В., Волобуев Д.М. Реконструкция аттракторов в трехмерном фазовом пространстве и моделирование геомагнитных пульсаций. // Вопросы геофизики. СПб: Изд. С.-Петербургского унив., 1998. В. 35. С. 338–348.
7. Zbilut J.P., Webber Jr.C.L. Embeddings and delays as derived from quantification of recurrence plots. // Physics Letters A 171 (3–4). 1992. P. 199–203.
8. Iwanski J.S., Bradley E. Recurrence plots of experimental data: To embed or not to embed? // Chaos. 8 (4). 1998. P. 861–871.
9. Webber Jr.C.L. Recurrence Quantification Analysis, 2003. URL: http://homepages.luc.edu/~cwebber
10. Киселев Б.В., Киселев В.Б. Различия в динамике солнечной и геомагнитной активности. // Вопросы геофизики. Вып. 39 СПб: Изд-во С.-Петербургского ун-та, 2006.
11. Вальчук Т.Е., Фельдштейн Я.И. Корреляционные и регрессионные соотношения между индексом аа геомагнитной активности и характеристиками околоземного космического пространства. / Солнечный ветер, магнитосфера и геомагнитное поле. М., Наука, 1983. 127 с.
12. http://www.krugosvet.ru/articles/110/1011048/1011048a2.htm

# ОБЗОР ТЕХНОЛОГИЙ ДОСТУПА К ДАННЫМ ДЛЯ ОПЕРАЦИОННОЙ СИСТЕМЫ WINDOWS

**В.А. Козак**
**Научный руководитель – А.А. Малинин**

В работе производится обзор наиболее весомых технологий доступа к данным для операционной системы Windows. Описаны их архитектурные модели и составные компоненты, выявлены достоинства и недостатки. Сделаны выводы, позволяющие судить об эффективности рассмотренных технологий и целесообразности их применения в дальнейшем.

## Введение

На сегодняшний день разработаны сотни баз данных различных типов, способных работать на компьютерах любой мощности – от мэйнфреймов до карманных компьютеров и других портативных мобильных устройств [1]. Кроме того, с каждым годом компьютеры одной организации или нескольких различных организаций все чаще соединяются друг с другом. Поэтому возникает необходимость в налаживании совместного доступа к базам данных по сети. Главным препятствием для совместного использования баз данных является несовместимость системного программного обеспечения и приложений, работающих на разных компьютерах.

С середины 1980-х годов программисты пытаются создать универсальный программный интерфейс API, который позволил бы им разрабатывать приложения, одинаковым образом взаимодействующие с различными источниками данных. За истекшее время было предложено множество решений в этой области. В данной работе производится обзор наиболее весомых технологий доступа к данным для операционной системы Windows. Большое внимание уделяется рассмотрению их архитектур и составных компонентов, выявляются достоинства и недостатки, а также описываются возможности доступа рассматриваемых технологий к различным источникам данных. К источникам могут относиться, например, различные базы данных, индексно-последовательный файл (ISAM file), текстовый файл (например, в формате XML) и т.д. Определяются возможности интеграции локальных и Web-приложений с базами данных. Представлена информация о том, какие технологии лучше использовать для локальных данных, а какие – для удаленных. Одним из основополагающих предыдущих исследований в данной области является статья Аша Рофаляя (Ash Rofail) и Яссера Шоуда (Yasser Shohoud) «VS 6.0 Benchmarks: New Features Don’t Impact Speed», опубликованная в журнале VBPJ [2]. Кроме того, существует еще ряд довольно мощных работ, но в большинстве из них рассматриваются две-три технологии. В данном же обзоре освещены все значительные разработки в данной области, включая последнюю технологию компании Microsoft – ADO.NET.

## Основная часть

Наиболее значительным среди ранних решений является ODBC (Open DataBase Connectivity, открытый интерфейс доступа к базам данных) – это стандартный интерфейс между базой данных и приложением, взаимодействующим с ней.

Первая версия Microsoft ODBC вышла в свет в 1992 году. Сейчас повсеместно используется третья версия ODBC, которая была представлена в 1996 году. ODBC выполнен в виде набора функций. Этот интерфейс создавался для доступа к реляционным базам данных. Сейчас же это единый универсальный интерфейс, позволяющий приложениям работать с СУБД различных типов, для которых имеется драйвер ODBC. Менеджер драйверов (Driver Manager) взаимодействует с приложением и обеспечивает загрузку драйвера, необходимого для доступа к конкретному источнику данных.

Используя ODBC, программист может не заботиться о деталях внутреннего устройства и особенностях естественного интерфейса различных СУБД, так как драйвер ODBC полностью скроет от него эти детали. В результате программы, обращающиеся к базам данных, становятся менее зависимыми от этих баз данных**.** Для того чтобы выполнить какое-либо действие с базой данных, необходимо использовать соответствующую команду SQL в качестве аргумента функции ODBC [3].

Интерфейс ODBC состоит из четырех функциональных компонентов, отраженных на рис. 1.

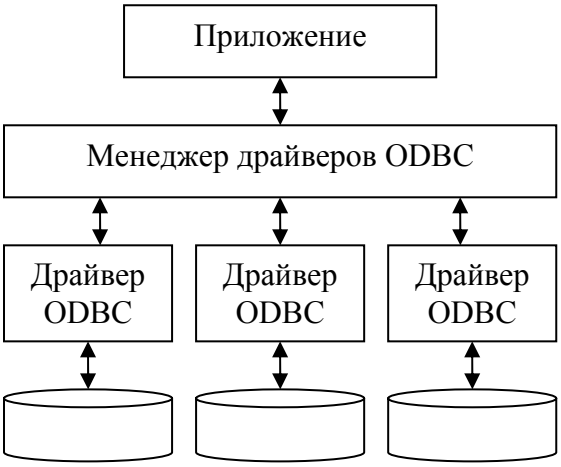



Рис. 1. Компоненты ODBC [3]

Благодаря каждому из них достигается гибкость ODBC, позволяющая взаимодействовать любым ODBC-совместимым клиентам и серверам. Ниже представлено описание этих компонентов.

- Приложение – часть ODBC, с которой непосредственно работает пользователь.
- Менеджер драйверов – библиотека динамической компоновки (dinamic link library, DLL), которая загружает необходимый драйвер.
- Драйвер – программный модуль, получающий вызовы функций посредством стандартного интерфейса ODBC и переводящий их в код, понятный источнику данных. Как только источник данных возвращает результат, драйвер в обратном порядке преобразует его в стандартный для ODBC вид.
- Источники данных – различные базы данных, индексно-последовательные файлы (ISAM file), текстовые файлы (например, в формате XML) и т.д. [3].

Неоспоримыми достоинствами ODBC являются его быстродействие, мощность, гибкость и переносимость исходного кода. Главный недостаток – сложность использования.

Проблема ODBC в его структуре – это просто набор функций. Такая структура подходит для процедурных языков, таких как C, но не соответствует принципам объектно-ориентированного программирования. Сегодняшние прикладные программисты привыкли работать с объектами, и на смену программным интерфейсам пришли объектные. В конце 1992 – начале 1993 года появились DAO (Data Access Objects) и RDO (Remote Data Objects). Появление двух разных механизмов было связано с необходимостью оптимизации решения двух отдельных задач: доступа к локальным и удаленным базам данных соответственно.

DAO (объекты для доступа к данным) – объектно-ориентированная технология доступа к данным компании Microsoft. DAO 1.0 появилась в ноябре 1992 года как API к системе баз данных Jet. Технология Jet поддерживала доступ к файлам формата MDB (Microsoft Access) и к источникам данных ISAM. Начиная с версии 3.1, появилась возможность использовать API DAO, не используя при этом язык БД Jet. В DAO 3.5 были

реализованы уже две объектных модели, первая работала с базами данных Microsoft Jet, вторая, названная ODBCDirect, использовала ODBC [4]. Чистый ODBC, безусловно, быстрее второй модели технологии DAO, но в использовании он гораздо сложнее. Кроме того, в библиотеке MFC (Microsoft Foundation Classes) для C++, библиотеках Visual Basic и в других языках существуют шаблоны, существенно упрощающие процесс разработки. Тип выбранной модели задается при создании соединения (рабочей области) указанием констант: dbUseJet – использовать Jet; dbUseODBC – использовать ODBC. Иерархия объектов DAO для модели Jet представлена на рис. 2, для модели ODBC – на рис. 3.

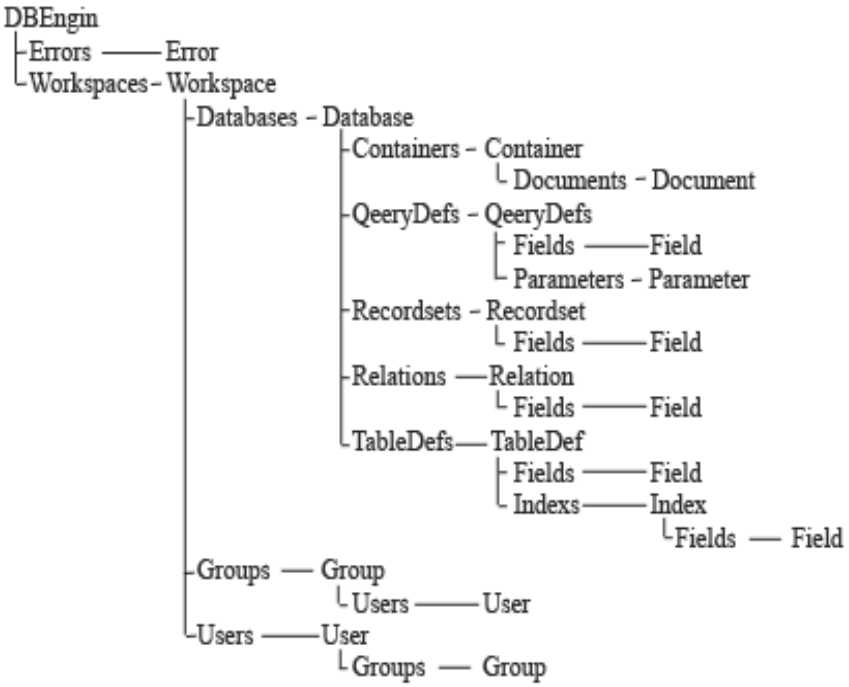



Рис. 2. Иерархия объектов DAO, модель Jet [4]

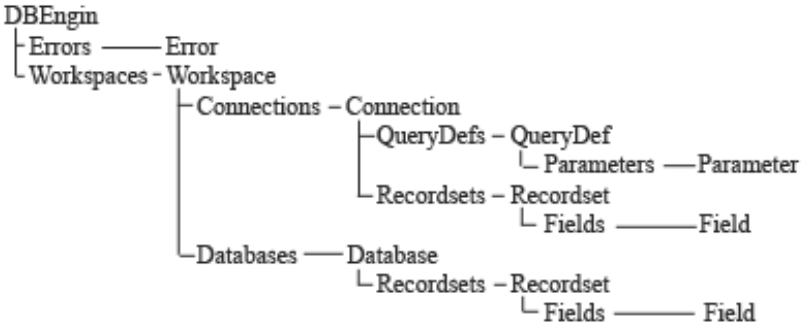



Рис. 3. Иерархия объектов DAO, модель ODBCDirect [4]

DAO исторически было ориентирована на обработку локальных данных, поддержка клиент-серверных источников данных, хотя и была реализована, не была достаточно эффективной. Для работы с удаленными базами данных использовался RDO. Доступ к локальным базам данных Jet и ISAM выполняется в RDO через драйверы ODBC, что снижает его быстродействие по сравнению с DAO. Однако RDO обеспечивает возможность работы с большим числом БД различных разработчиков, предоставляя средства доступа к таким элементам, как, например, запоминаемые процедуры и сложные результирующие наборы данных.

Однако естественное развитие вычислительных систем привело к необходимости создания единого механизма, который обеспечил бы единый подход при работе

с БД различных классов. И в середине 1990-х годов, с развитием и распространением технологии COM (Component Object Model), компания Microsoft объявила о постепенном переходе к использованию новой технологии OLE DB (Objects Linking and Embedding – объекты связанные и внедренные). Объектный интерфейс OLE DB представляет собой открытый стандарт, предназначенный для универсального доступа приложений к базам данных.

OLE DB – это совокупность интерфейсов ActiveX, которые упрощают и унифицируют доступ к данным независимо от того, где и как они хранятся. Приложение, использующее эти элементы, может интегрировать информацию из СУБД (SQL Server, ORACLE, Access) с информацией из систем другого типа, но поддерживающих OLE-механизм [3]. Технология позволяет одинаковым образом обращаться к реляционным базам данных, серверам почты, базам данных для мэйнфреймов с методами доступа IMS, VSAM и т.д. [5].

Непосредственный доступ к источникам данных обеспечивают провайдеры (provider) OLE DB. Поэтому интерфейсы поддерживают тот объем функциональных возможностей систем управления базами данных, который соответствует конкретному источнику данных. Наиболее часто используемые провайдеры приведены в таблице.

| **Драйвер** | **Провайдер** | **Описание** |
|---|---|---|
| MSDASQL | ODBC Drivers | Драйверы ODBC (по умолчанию) |
| Microsoft.Jet.OLEDB.3.5 | Jet 3.5 | Только базы данных MS Access 97 |
| Microsoft.Jet.OLEDB.4.0 | Jet 4.0 | Базы данных MS Access и другие БД |
| SQLOLEDB | SQL Server | Базы данных MS SQL Server |
| MSDAORA | Oracle | Базы данных Oracle |
| MSDAOSP | Simple provider | Для создания ваших собственных провайдеров, для простых текстовых данных |

Таблица. Наиболее часто используемые провайдеры OLE DB [6]

Спецификация компонентов OLE DB представлена на рис. 4.

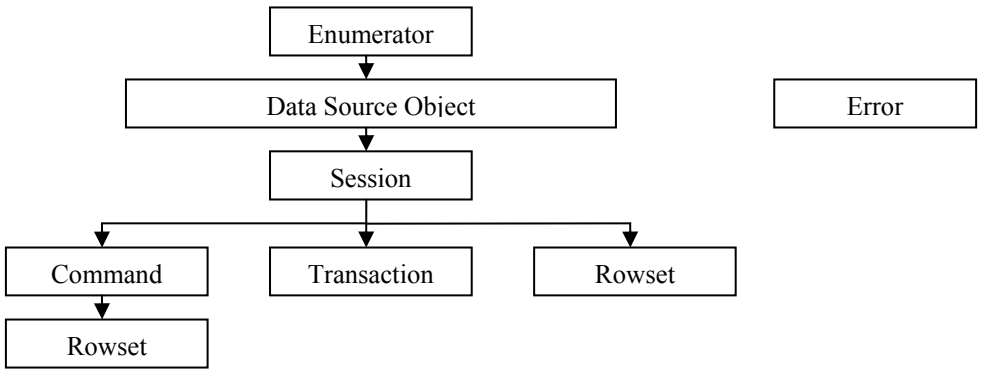



Рис. 4. Модель компонентов OLE DB [3]

Однако OLE DB, по мнению самой компании Microsoft, является интерфейсом системного уровня, этот интерфейс должен использоваться системными программистами. Технология OLE DB является тяжеловесной, сложной и очень чувствительной к ошибкам. Чтобы облегчить работу с OLE DB, в 1996 году был создан дополнительный прикладной объектный интерфейс более высокого уровня, который получил название ADO (ActiveX Data Objects). Ключевыми элементами этой модели является набор объектов, с помощью которых можно:

- создать соединение с базой данных;
- выполнить команду с параметрами;

- получить результаты выполнения этой команды в виде переменной или набора записей;
- обработать события или ошибки [1].

Отсутствие сложной структуры делает его более гибким и легким в употреблении. Модель объектов ADO представлена на рис. 5.

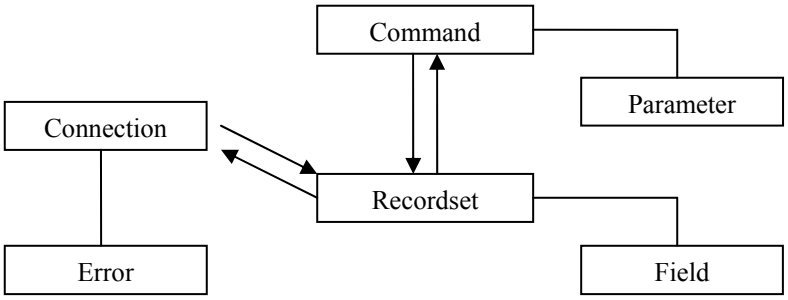



Рис. 5. Модель компонентов ADO [4]

Здесь стоит упомянуть о результатах тестирования производительности разных методов доступа к данным, приведенных в статье «VS 6.0 Benchmarks: New Features Don't Impact Speed» в журнале VBPJ. Там говорится об исследовании скоростных характеристик разных средств, входящих в состав Visual Studio 6.0, а также VB 6, для различных режимов работы программ (графика, математические задачи, обработка строк и пр.), в том числе и при работе с базами данных.

По результатам тестирования при работе с удаленными данными быстродействие ADO и RDO примерно одинаково. Что же касается локальных баз данных, то скорость ADO в 3–5 раз ниже, чем у DAO [2]. Это плата за универсальность, такой механизм всегда уступает специализированным, имея в качестве преимущества упрощение процесса разработки. Современный стиль разработки заключается именно в таких приоритетах: главное – сократить трудоемкость разработки, а уже потом думать о повышении эффективности приложений. Более важным является другой момент: Microsoft тогда объявила, что будет усовершенствовать и обновлять только модель ADO (что соответствует общей стратегии корпорации на унификацию разных технологий на базе ActiveX), т.е. DAO и RDO постепенно должны были сойти со сцены. Результаты тестирования для локальных данных отражены на рис. 6, для удаленных – на рис. 7.

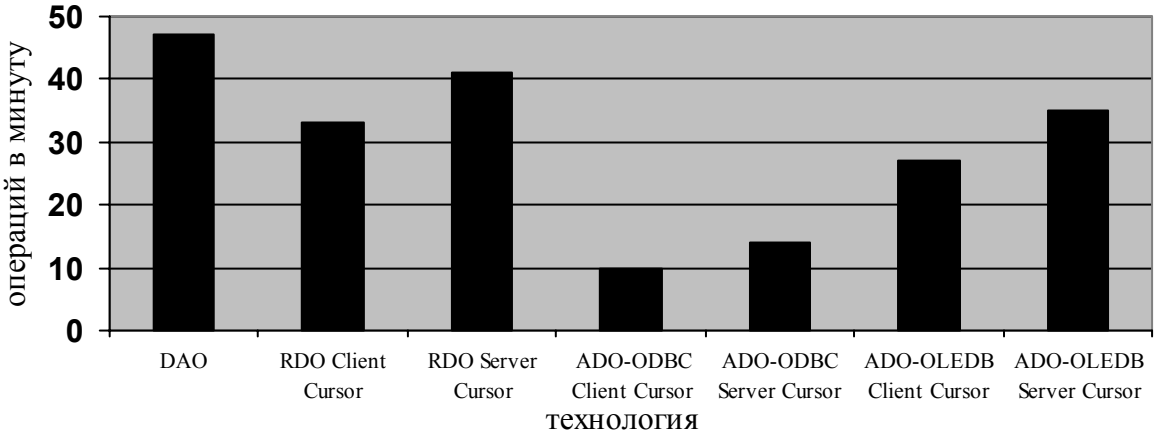



Рис. 6. Скорость доступа к локальным данным [2]

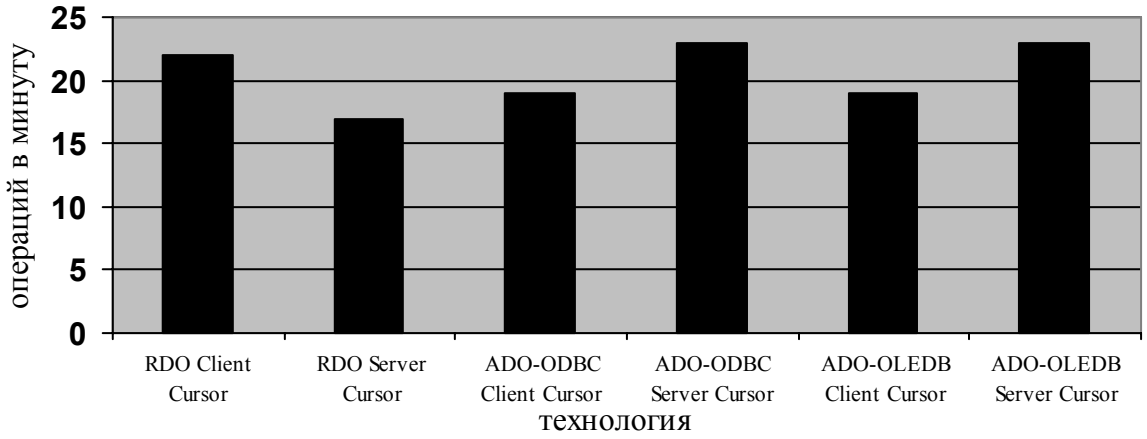



Рис. 7. Скорость доступа к удаленным данным [2]

Но сегодня и ADO доживает свои последние дни. В 2003 году компания Microsoft анонсировала выход своей новой платформы: .Net Framework, которая является очередной попыткой компании Microsoft заново перепроектировать средства и инструменты разработки программного обеспечения, чтобы сделать их более удобными для создания веб-приложений. ADO.NET является новым развитием ADO, ориентированным на решение проблем, связанных с разработкой веб-систем, и устраняющим многие недостатки устаревшей технологии ADO. Проблема ADO состоит в том, что эта технология основана на COM. Для одно- и двухзвенных приложений COM является вполне приемлемой платформой, однако в мире Веб использовать COM в качестве транспортного механизма фактически невозможно. Для COM характерны три основные проблемы, которые ограничивают использование этой технологии в Веб: во-первых, COM функционирует только в среде Windows, во-вторых, передача наборов записей требует маршализации COM, в-третьих, вызовы COM не могут проникать через корпоративные брандмауэры. Технология ADO.NET решает все три проблемы благодаря использованию XML [5]. Совместная работа классов ADO.NET и XML в .NET Framework обеспечивается объектом DataSet. Это очень мощный объект, обладающий встроенным форматом сериализации XML, с помощью которого можно легко и надежно передавать данные.

Другой важной особенностью ADO.NET является использование разъединенной модели доступа к данным. В клиент-серверных приложениях традиционно используется технология доступа к источнику данных, при которой соединение с базой поддерживается постоянно. Однако после широкого распространения приложений, ориентированных на Интернет, выявились некоторые недостатки такого подхода. ADO.NET предлагает следующую схему взаимодействия клиента с сервером:

- открытие соединения с сервером СУБД;
- отправка запроса к базе данных;
- закрытие соединения;
- обработка данных, полученных в виде объекта класса DataSet;
- открытие соединения с сервером СУБД;
- обновление базы данных с использованием содержимого объекта класса DataSet;
- закрытие соединения [1].

По количеству потребляемых ресурсов и времени выполнения одной из самых затратных команд является выборка данных. Желая максимально повысить скорость считывания данных из БД, программисты Microsoft пришли к тому, что вся выборка кэшируется на стороне клиента в момент создания DataReader'а [8]. Такое поведение плохо скажется на скорости при частых обращениях к серверу баз данных. Однако описанная

выше разъединенная модель доступа к данным для ADO.NET показывает, что основную работу пользователь будет производить на локальной машине. В этом случае клиентское приложение один раз выкачивает данные из базы данных и, используя сохраненные в кеше данные, с максимальным быстродействием формирует объект DataSet – прообраз базы данных на клиенте.

Таким образом, автор видит технологию ADO.NET достаточно универсальной, логичной и несложной в использовании, отвечающей современным требованиям для распределенных приложений (именно для таких и создавался .NET). Кроме того, при использовании провайдеров, «заточенных» под конкретную базу данных, данная технология демонстрирует быстродействие, сопоставимое с ODBC. Если же применять универсальные провайдеры, такие как ODBC.NET Data Provider или OLE DB.NET Data Provider, быстродействие резко падает – это главный недостаток данной технологии. Тем не менее, именно ADO.NET в ближайшее время будет наиболее популярной технологией для доступа к данным в операционной системе Windows.

Архитектура ADO.NET представлена на рис. 8.

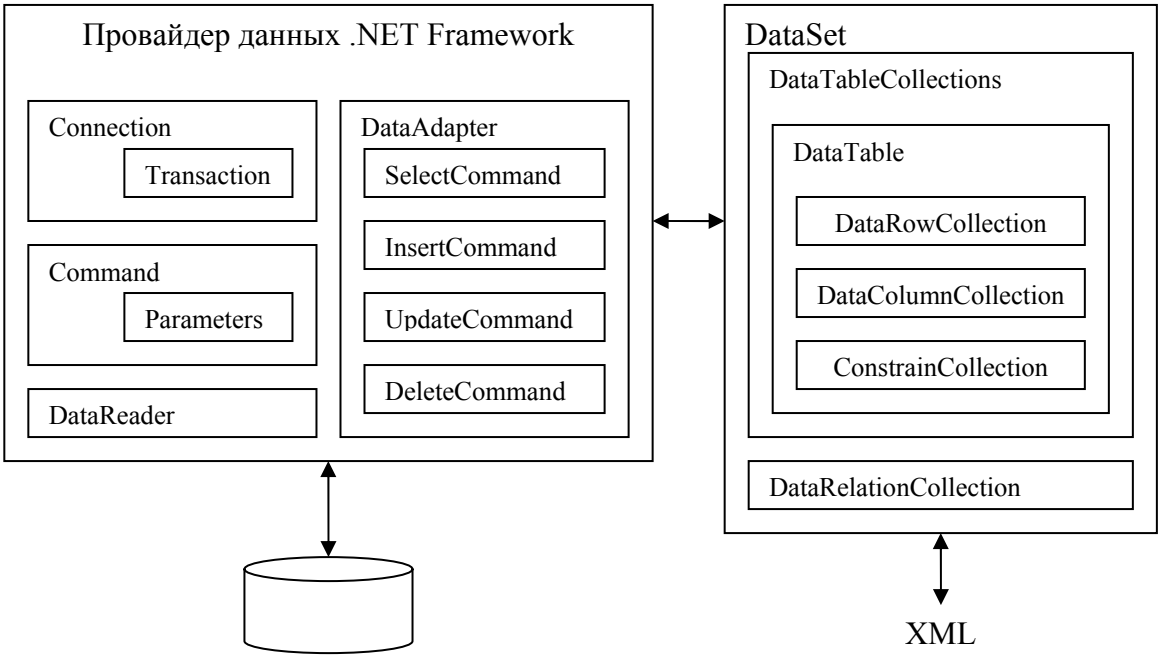



Рис. 8. Архитектура ADO.NET

**Заключение**

В работе проведен обзор основных технологий доступа к данным для операционной системы Windows. Оценивая эволюцию этих технологий, можно сделать вывод, что основной задачей при их разработке было: независимо от типа создаваемого приложения (многоуровневый, распределенный, автономный или Web-ориентированный клиент/сервер) разработчик должен иметь «наилучший» набор инструментов, приложений и источников данных, позволяющих создать такое приложение. Таким образом, для выбранных технологий оценивалось их быстродействие, выявлялись возможности доступа к данным различной природы, изучались способности работы в распределенных системах. В итоге можно сделать вывод, что сегодня мы обладаем технологиями, которые позволяют в кротчайшие сроки реализовывать довольно сложные проекты. Среди решений Microsoft к ним, в первую очередь, относится ADO.NET. Эта технология – не самая быстрая в работе (при использовании стандартных провайдеров), зато она более универсальна, а интуитивно понятные абстракции ADO.NET более просты в использовании и изучении.

## Литература


1. Фролов А.В., Фролов Г.В. Визуальное проектирование приложений C#. М.: Кудиц – образ, 2003. 512 с.
2. Rofail A., Shohoud Y. VS 6.0 Benchmarks: New Features Don't Impact Speed. // VBPJ, 1998. № 14. С. 30–44.
3. Пирогов В.Ю. Ms SQL Server 2000 управление и программирование. СПб: БХВ - Петербург, 2005. 608 с.
4. Арчер Т., Уайтчепел Э. Visual C++.NET. Библия пользователя. М-СПб-К.: Вильямс, 2005. 1216 с.
5. Троэлсен Э. C# и платформа.NET. СПб: Питер, 2006. 796 с.
6. Кэнту М. Delphi 7. Для профессионалов. СПб: Питер, 2003. 840 с.
7. Марков А.С., Лисовский К.Ю. Базы данных. М.: Финансы и кредит, 2006. 512 с.
8. Михаилов С. Сравнение скорости доступа к данным (ADO.NET, ADO, ascDB). // RSDN Magazine. 2003. № 3. С. 3–34.

# ИССЛЕДОВАНИЕ ТЕХНОЛОГИЙ ДОСТУПА К ДАННЫМ ДЛЯ ОПЕРАЦИОННОЙ СИСТЕМЫ WINDOWS

**В.А. Козак**
**Научный руководитель – А.А. Малинин**

Эффективный доступ к данным – эту задачу приходится решать в любом крупном проекте. Неотъемлемым атрибутом эффективности является быстродействие. В работе производится исследование наиболее весомых технологий доступа к данным для операционной системы Windows на предмет быстродействия. При помощи ряда тестовых приложений выделены фавориты, сделаны попытки объяснить полученные результаты.

## Введение

В сложных распределенных системах очень важно поддерживать актуальность используемых данных, поэтому мы должны уметь их быстро изменять. Кроме того, именно недостаточная скорость выполнения приложения вынудит пользователя отказаться от использования такого продукта. Об архитектурных особенностях различных технологий сказано уже немало, а целью данной работы было выявить технологию с лучшей производительностью. Для выполнения такой задачи был разработан набор тестовых программ, осуществляющих подключение к базе данных и выполнение несложных запросов. Поскольку тестировались не сами базы данных, а средства доступа к ним, то применение изощренных запросов видится автору нецелесообразным. Среди предыдущих работ в данной области хочется отметить статью С. Михайлова «Сравнение скорости доступа к данным (ADO.NET, ADO, ascDB)», опубликованную в журнале «RSDN Magazine» в 2003 году [1]. Но в ней автор использует другую базу данных, поэтому результаты выполнения аналогичных запросов несколько отличаются. Кроме того, нужно учесть, что в той работе среди исследуемых технологий был их собственный продукт. В данном исследовании набор технологий был дополнен ODBC, а для ADO.NET рассматривались два провайдера: стандартный ODBC Provider и специализированный MySQL Provider. Также автор хотел бы поделиться своим субъективным мнением по поводу удобства использования той или иной технологии.

## Описание тестов

Для тестирования были отобраны следующие технологии: ODBC, ADO, ADO.NET (ODBC Provider и MySQL Provider). По различиям в этих технологиях можно судить о развитии программирования под Windows. ODBC представляет собой набор динамически подключаемых библиотек [2]. ADO основано на COM, эта модель стала в свое время целой эпохой в развитии клиент/серверных приложений на базе Windows [3]. ADO.NET – часть новой глобальной платформы от Microsoft. Клиентские приложения, использующие ADO.NET, написаны на C#, остальные – на C++. В качестве сервера баз данных использовался MySQL 5.0.20. Сервер был установлен на локальную машину из бинарного дистрибутива, все параметры приняты «по умолчанию». В нем были созданы тестовые таблицы, содержащие 9 полей со следующими типами данных: строки с длиной от 10 до 100 символов (5 полей), int (3 поля), double (одно поле). Таблицы создавались в отдельной базе данных и не связаны с другими таблицами или между собой.

Выполнялись следующие тесты.

1. «ForwardAll» – получение аналога forward-only курсора на 1 миллион записей с прокруткой на различное количество записей. Курсор Forward-only (только вперед) позволяет вам перемещаться по набору данных в направлении от начала к концу [4].

2. «Select100» – 1000 раз получение аналога forward-only курсора на 100 записей с прокруткой всего списка.
3. «Static» – получение аналога static-курсора на 100 тысяч записей и прокрутка его 1000 раз на удаленные друг от друга позиции (1000 записей с возвратом на первую запись). При использовании статического курсора набор данных полностью перемещается на сторону клиента [4].
4. «ManyConnection» – 1000 раз добавление (изменение, удаление) одной записи. Для каждой операции создавалось новое соединение.
5. «OneConnection» – проведение 1000 операций по добавлению, изменению, удалению одной записи, используя единственное соединение с базой данных.

Каждый тест выполнялся в «in-process» режиме, а ADO.NET тестировалось также и в локальном режиме через COM+. В первом случае весь исполняемый код находился в одном exe-приложении, т.е. приложение напрямую работало с базой данных, минуя промежуточное копирование данных и передачу их между процессами. Во втором – работа велась через отдельный промежуточный COM-объект, создаваемый в другом процессе. В этом объекте происходило получение result-set'a, данные из которого затем возвращались клиентскому приложению. Для передачи данных между процессами использовался объект DataSet.

В качестве тестовой машины использовался компьютер со следующими параметрами:

- ОС Microsoft Windows XP Professional Service Pack 2;
- AMD Athlon(tm) 64 Processor 3000+, 2010 MHz;
- RAM 512 Мб.

### Особенности теста ForwardAll

В этом тесте операции проводятся над очень большой таблицей в forward-only режиме. На базу данных посылался запрос "SELECT * FROM EmployeesMySQL", в таблице миллион записей. Цель этого теста – проверить, станет ли MySQL сразу производить выборку или будет искать нужные строки во время прокрутки курсора. Такое поведение демонстрирует, например, MS SQL Server. Как оказалось, MySQL честно подготовил выборку сразу после посылки команды «выполнить», что не лучшим образом сказалось на времени выполнения теста. В среднем около 11 секунд тестовые программы ждали, пока сервер закончит выборку. Далее из полученной выборки на клиентское приложение извлекались отдельные строки – производился перебор выборки. В зависимости от количества перебираемых строк мы получали различное время. Было интересно, каково же отношение скорости формирования выборки на сервере к скорости обработки записей из этой выборки и как такое поведение MySQL сервера скажется на быстродействии при обработке части первоначальной выборки.

Код программы этого теста для технологии ODBC был написан на C++ без использования каких-либо вспомогательных классов из библиотеки MFC или других. Производился непосредственный вызов функций ODBC, которым передавались необходимые параметры. Таким образом, код оказался довольно-таки громоздким. Начальные установки в ODBC удовлетворяют условиям нашего теста. При выделении памяти для идентификаторов использовалась функция SQLAllocHandle, для выполнения подготовленного оператора – SQLExecute, для перемещения по выборке – SQLFetch. Чтобы в дальнейшем не возвращаться к написанию кода для базовых операций, ниже представлена последовательность вызова основных функций. Каждая ODBC функция возвращает код успеха или неудачи, для упрощения восприятия в представленном коде эти результаты не проверяются и никак не обрабатываются. На самом деле код этого теста для ODBC занял более ста строк.

```
SQLHENV henv = NULL;                    /* дескриптор среды */
SQLHDBC hdbc = NULL;                    /* дескриптор подклчения к БД */
SQLHSTMT hstmt = NULL;                  /* дескриптор оператора */
SQLRETURN rc;                           /* результат выполнения команды*/
SDWORD sDWord;                          /* размер считываемого столбца */
int employeeId;                 /* сюда запишем одно считанное значение для столбца типа int */
//получаем уникальный дескриптор среды
rc = ::SQLAllocHandle(SQL_HANDLE_ENV, SQL_NULL_HANDLE, &henv);
//устанавливаем версию ODBC
rc = :: SQLSetEnvAttr(henv, SQL_ATTR_ODBC_VERSION, (SQLPOINTER)SQL_OV_ODBC3,
SQL_IS_INTEGER);
//получаем дескриптор подклчения к БД
rc = ::SQLAllocHandle(SQL_HANDLE_DBC, henv, &hdbc);
//подключаемся к источнику данных
rc = ::SQLConnect(hdbc, (SQLCHAR*)"MySQLdata", SQL_NTS,
        (SQLCHAR*)"user", SQL_NTS, (SQLCHAR*)"password", SQL_NTS);
//получаем дескриптор оператора
rc = ::SQLAllocHandle(SQL_HANDLE_STMT, hdbc, &hstmt);
//текст запроса
LPCSTR szSQL = "SELECT * FROM EmployeesMySQL";
//подготавливаем оператор
rc = ::SQLPrepare(hstmt, (unsigned char*)szSQL, SQL_NTS );
//выполнение
rc = ::SQLExecute(hstmt);
//перебор по выборке (прокрутка countRow) строк
for(int i = 0; i<countRow; i++)
{
        //считываем данные из курсора. Здесь пример для одного столбца
        SQLBindCol(hstmt, 1, SQL_C_LONG, & employeeId, 0, &sDWord);
        //перемещение курсора
        rc = SQLFetch(hstmt);
}
//далее необходимо освободить выделенную память, закрыть курсор и соединение
```

В тесте «ForwardAll» для ADO в качестве forward-only курсора использовался объект Recordset с установкой значений свойств: CursorType в adOpenForwardOnly, LockType в adLockReADOnly и CursorLocation в adUseServer. Переход по записям осуществлялся с помощью вызова метода MoveNext. В результате перебор в ADO оказался на 30 % менее эффективен, чем в ODBC. Часть кода этого теста для ADO представлена ниже.

```
CoInitialize(NULL);
HRESULT hr;
int employeeId;
_ConnectionPtr myConnection;
myConnection.CreateInstance(__uuidof(Connection));
//открытие соединения
hr = myConnection->Open("DSN=MySQLdata;", _bstr_t("user"), _bstr_t("password"), adModeUnknown);
_CommandPtr pCommand;
pCommand.CreateInstance(__uuidof(Command));
pCommand->ActiveConnection = myConnection;
pCommand->CommandText = "SELECT * FROM EmployeesMySQL";
_RecordsetPtr pRecordset;
pRecordset.CreateInstance(__uuidof(Recordset));
pRecordset->CursorLocation = adUseServer;
//открытее Recordset'а, сейчас команда посылается на СУБД
pRecordset->Open((IDispatch*)pCommand, vtMissing, adOpenForwardOnly,  adLockBatchOptimistic,
adCmdUnknown);
// далее приведены действия, необходимые для чтения одного столбца
FieldsPtr pFields = pRecordset->Fields;
FieldPtr pEmpId = pFields->GetItem("EmployeeID");
ASSERT(NULL != pEmpId);
```

```
_variant_t vEmpId;
//прокрутка Recordset'а на countRow строк
for(int i=0; i< countRow; i++)
{
        vEmpId = pEmpId->Value;
        employeeId = (int) vEmpId;
        pRecordset->MoveNext();
}
pCommand->Cancel();
myConnection->Close();
CoUninitialize();
```

В качестве аналога forward-only курсора в ADO.NET использовался IDataReader. Считывание производилось по одной записи, использовался метод Read. Как выяснилось, данная технология открывает соединение гораздо быстрее предыдущих. Когда же дело доходит до чтения данных, становится понятно, зачем разработчики баз данных создают специализированные провайдеры. В отличие от ODBC Provider'а, вынужденного взаимодействовать с менеджером драйверов, загружать конкретный драйвер, подключаться к базе данных по ее DNC, специализированный провайдер взаимодействует с базой данных напрямую. В итоге: ADO.NET с ODBC Provider демонстрирует самые плохие результаты перебора среди тестируемых технологий, тогда как ADO.NET с MySQL Provider – лучшие. Ниже представлен код для MySQL Provider'а.

```
int employeeId;
MySqlConnection connection = null;
IDataReader reader = null;
try {
        connection = new MySqlConnection(connectionString);
        MySqlCommand command = connection.CreateCommand();
        connection.Open();
        command.CommandText = "SELECT * FROM EmployeesMySQL";
        reader = command.ExecuteReader();
        for(int j = 0; j < countRow; ++j) {
                // Переходить на след. запись с чтением данных
                reader.Read();
                employeeId = reader.GetInt32(0);
        }
        reader.Close();
        connection.Close();
} catch(MySqlException exc) {
        if(connection != null)
                connection.Close();
        if(reader != null)
                reader.Close();
}
```

В случаях, когда возвращаемая выборка имеет такой большой размер, очень полезно иметь возможность считать небольшую часть выборки и сразу же начать ее использовать, а в дальнейшем производить быструю подкачку данных из выборки. ODBC и ADO реализуют такой механизм путем установки размера кэша курсора. В ADO.NET для считывания одновременно нескольких записей подходит лишь объект DbDataAdapter. Действительно, используя метод Fill, в параметры которого можно передать номер стартовой записи и количество считываемых записей, можно очень эффективно произвести первое считывание. Но, когда в следующий раз мы вызываем метод Fill, происходит переоткрытие IDataReader'а, и мы вновь ждем, пока сервер произведет выборку миллиона строк. Следовательно, на получение следующей порции данных в нашем случае уйдет минимум 11 секунд. Такой результат неприемлем.

При использовании IDataReader'а напрямую, как в нашем случае, возникают другие проблемы. В этом объекте нет возможности пропускать ненужные строки и переходить к началу выборки. А поскольку на одном соединении с базой данных может

быть открыт только один IDataReader, то, пропустив одну лишь строку в переборе, вы уже не сможете к ней вернуться.

Итоги теста ForwardAll отражены на рис 1.

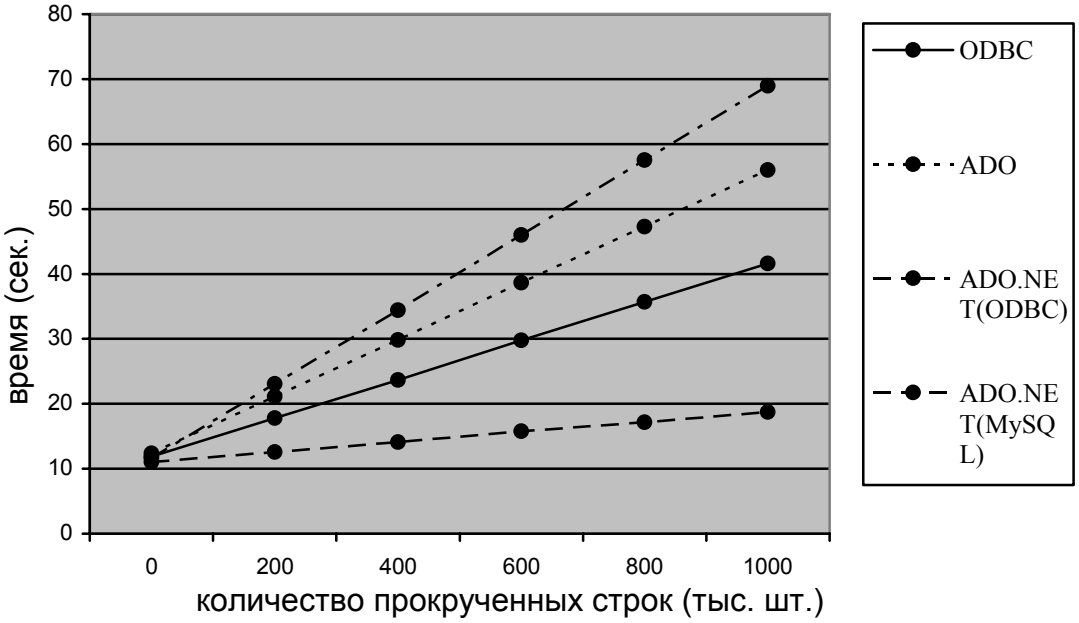



Рис. 1. Зависимость времени выборки от количества перебираемых строк в «in process» режиме

**Особенности теста Select100**

Этот тест был проведен с целью подтверждения «виновности» MySQL сервера в потере 11 секунд в тесте ForwardAll. Здесь мы также получаем forward-only курсор, только уже на 100 записей, и прокручиваем весь список. Приведенное в табл. 1 время соответствует 1000 повторов в цикле этого теста. В итоге выборка и перебор ста записей (одна итерация цикла) заняла, в зависимости от технологии, 3–9 миллисекунды. Следовательно, небольшую выборку сервер подготовил практически мгновенно, а трехкратный разброс скорости перебора только подтвердил результаты предыдущего теста.

| | ODBC | ADO | ADO.NET (ODBC) | ADO.NET (MySQL) |
|---|---|---|---|---|
| Select100 | 8.768 | 9.640 | 9.5153 | 2.776 |
| Static | 1.327/0.021 | 2.503/0.06 | 8.134/0.007 | 3.001/0.007 |
| Insert 1000 ManyConnection | 14.095 | 13.612 | 10.343 | 4.535 |
| Update 1000 ManyConnection | 14.037 | 12.783 | 10.287 | 4.495 |
| Delete 1000 ManyConnection | 14.107 | 12.469 | 10.375 | 4.549 |
| Insert 1000 OneConnection | 0.803 | 2.205 | Невозможно | Невозможно |
| Update 1000 OneConnection | 0.788 | 2.201 | Невозможно | Невозможно |
| Delete 1000 OneConnection | 0.812 | 2.249 | Невозможно | Невозможно |

Таблица 1. Быстродействие доступа к данным в «in process» режиме (с)

Итогом этих двух тестов стали следующие выводы.

- Несмотря на все рекламные заявления, MySQL непростительно долго готовит выборку большого размера.
- Для перебора ODBC оказалась наиболее быстрой из универсальных технологий.
- Специализированный провайдер MySQL во время перебора опережал ODBC в 4 раза. Это прекрасное средство для скачивания больших объемов информации.
- На ADO.NET невозможно организовать эффективную подкачку данных, что необходимо в случае такой большой выборки.

### Особенности теста Static

В данном тесте также хотелось поработать с таблицей на миллион записей, но с ADO.NET опять возникли проблемы. Объект DataSet, так разрекламированный во всех описаниях технологии, оказался неспособен обработать большую выборку. Проблема кроется во внутреннем механизме сериализации данных, реализованном для этого объекта. Если заполнение DataSet'а 200 000 записей происходит более-менее адекватно, то на 300 000 программа заметно тормозит. При увеличении выборки до 500 000 программа во время заполнения DataSeta'а (вызов метода Fill объекта DbDataAdapter) использовала до 450 Мб оперативной памяти и, загрузив процессор на 97 %, благополучно зависала.

В итоге, учитывая проблемы с DataSet, в тесте «Static» на сервер посылался запрос на выборку 100 000 записей. Далее 1000 раз производится его прокрутка на удаленные друг от друга позиции. При использовании статического курсора набор данных полностью перемещается на сторону клиента. Для этого в ODBC при помощи функции SQLSetStmtAttr атрибуту SQL_ATTR_CURSOR_TYPE было присвоено значение SQL_CURSOR_STATIC. Перемещение по выборке осуществлялось функцией SQLFetchScroll. Других принципиальных отличий от предыдущих примеров нет. В ADO объект Recordset создавался со следующими атрибутами: значение свойства CursorType – adOpenStatic, LockType в adLockReADOnly и CursorLocation в adUseClient. Переход по записям осуществлялся с помощью вызова метода Move. Для ADO.NET статический курсор создавался с использованием объекта DataSet, данные в который заносились объектом DbDataAdapter. Вначале данные помещались в этот объект, затем определенные строки извлекались, используя свойство Rows объекта Table, входящего в DataSet. Основной код для этого случая представлен ниже.

```
DataSet dataSet = null;
MySqlConnection connection = null;
IDataReader reader = null;
string commandText = "SELECT * FROM EmployeesMySQL WHERE (EmployeeID<=100000)";

try {
    connection = new MySqlConnection(connectionString);
    connection.Open();
    IDbDataAdapter my_adapter = new MySqlDataAdapter(commandText, connection);
    dataSet = new DataSet();
    my_adapter.Fill(dataSet, 0, 100000, "EmployeesMySQL");
    for(int i=0; i<100; i++) {
        DataRow row = dataSet.Tables[0].Rows[i*1000];
        employeeId = (int)row["EmployeeID"];
}
connection.Close();
} catch(MySqlException exc) {
    if(connection != null)
        connection.Close();
    if(reader != null)
        reader.Close();
}
```

Поскольку данные уже находятся на стороне клиента, доступ к ним осуществляется значительно быстрее, чем в случае forward-only курсора. В представленных в табл. 1 данных отдельно отражены как время выборки и кэширования на клиенте ста тысяч записей, так и время перебора. Здесь во время формирования выборки и передачи ее клиентскому потоку ODBC вновь предпочтительней. Низкая производительность ADO.NET связана с формированием DataSet'а. Зато доступ к полученным данным, когда они уже приведены к необходимому типу, осуществляется гораздо быстрее, чем в других технологиях.

### Особенности теста ManyConnection

В этом тесте на СУБД посылаются команды изменения данных. Производится вставка, изменение и удаление строк в тестовой таблице. Для каждой команды открывается свое соединение, после выполнения операции соединение закрывалось. Результаты, приведенные в табл. 1, соответствуют 1000 повторов вышеуказанных команд. Такое поведение сильно напоминает разъединенную модель, реализованную в ADO.NET, где одному соединению может соответствовать только одна команда.

Код для ODBC несложен, а про ADO надо сказать, что необходимую команду на сервер можно выполнить двумя способами: выполнить метод Execute интерфейса Command или вызвать метод Open интерфейса Recordset. Первый способ полезен, когда о результатах выполнения заботиться не нужно, второй необходим для возвращения набора записей из базы данных. В данном случае вызывался метод Execute. ADO.NET в этом тесте чувствовала себя наиболее комфортно, и результаты отражают это. В этой технологии очень быстро открывается соединение, а поскольку результаты команды обрабатывать не нужно, ADO.NET – лидер, причем, в случае с MySQL Provider'ом, превосходство над ODBC трех кратное.

### Особенности теста OneConnection

Этот тест отличается от предыдущего тем, что на сервер посылается 1000 команд изменения данных, используя единственное соединение. Использование ADO.NET в таком режиме невозможно. Для этой технологии можно создать локальный DataSet, произвести над ним необходимые команды и отправить весь набор данных на сервер разом. Но такое поведение не удовлетворяет условиям теста. Необходимо проведения 1000 команд изменения данных, а не одной – закачки DataSet'а. Таким образом, ADO.NET в этом тесте участвовать не будет. Но это не просто архитектурное ограничение последней технологии, отсутствие возможности такого поведения – огромный недостаток в случае, когда необходима немедленная доставка данных на сервер. Разница в быстродействии последних двух тестов очень велика, и она была бы гораздо больше при использовании удаленного сервера, поэтому отсутствие такого поведения в ADO.NET непонятно.

Проанализируем полученные результаты. Разница в быстродействии ODBC и ADO в тесте ManyConnection невелика. Но в тесте OneConnection ODBC обогнал конкурента в три раза. Отсюда вывод: в ODBC очень удачно реализована команда, отправляющая запрос на сервер, ADO же гораздо быстрее открывает соединение с последним.

На этом исследование технологий в «in-process» режиме завершено. Подведем некоторые промежуточные результаты. Разработанная в начале 90-х годов технология ODBC так и не потеряла своей актуальности. Ее неоспоримым достоинством является быстродействие. Среди недостатков, в первую очередь, выделяется сложность использования для современных прикладных программистов.

Технология ADO, ставшая заключительным этапом развития технологий для работы с данными, основанными на OLE и COM, уступает ODBC по производительности, но в ис-

пользовании ADO гораздо удобней и обладает большей функциональностью. Кроме того, действие, выполняемое в ODBC вызовом нескольких функций, в ADO часто может быть выполнено вызовом одной, более оптимизированной функции. Несмотря на это, ADO постепенно уходит из поля зрения программистов, причина этому – новая платформа .NET и входящая в ее состав технология ADO.NET. Эта технология успешно работает в распределенных системах и использует разъединенную модель доступа к данным [5]. Для данной технологии желательно использовать специализированные провайдеры. В этом случае быстродействие не будет уступать ODBC, а порой и превосходит последнюю. Если же применять универсальные провайдеры, такие как ODBC Provider или OLE DB Provider, быстродействие такой системы будет существенно уступать предыдущим. Кроме того, количество кода, которое должен написать программист для использования ADO.NET, несравнимо меньше, чем в случае с какой-либо другой технологией.

## COM+

С использованием технологии COM+ тестировалось только ADO.NET. Основной целью здесь была проверка эффективности взаимодействия модели компонентных объектов и .Net в целом. Это актуально, поскольку очень часто сборки .NET должны успешно работать в мире сложных приложений, где значительную часть кода составляют классические COM-серверы. Помимо классического COM, технология COM+ объединила в себе возможности MTS (Microsoft Transaction Server) – очень мощного сервера приложений для размещения приложений уровня предприятия. Кроме того, COM+ добавила многие важные возможности и устранила недостатки этих двух систем [6].

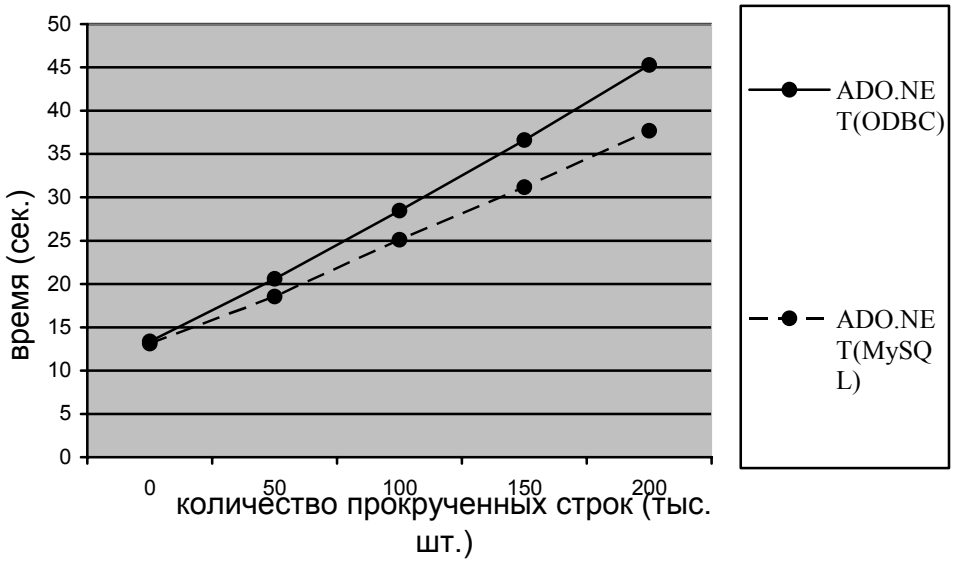



Рис. 2. Зависимость времени выборки от количества перебираемых строк в COM+ режиме

Для возвращения результатов работы COM+ объекта в основной поток использовался объект DataSet. Код серверной части, непосредственно обращающийся к базе данных, ничем не отличается от предыдущего. Но сами классы и методы, содержащие этот код, имеют много дополнительных атрибутов из пространства имен System.EnterpriseServices. Кроме того, клиент будет взаимодействовать не с самим классом, содержащим выполняемый код, а с интерфейсом, имплементируемым этим классом. Вообще COM – очень сложная технология, и она не является темой данной работы. Поэтому о реализации тестов, взаимодействующих с COM, хочется сказать лишь следующее: помимо компиляции кодов серверной части, для удобного взаимодействия с ними клиента полученный файл

(Server.dll) был зарегистрирован в GAC (global assembly cache) утилитой gacutil.exe. После этого объекты из библиотеки Server.dll были зарегистрированы в COM+ при помощи утилиты regsvcs. Затем было написано клиентское приложение, вызывающее методы серверного. Клиент использовал позднее связывание, т.е. получал информацию о типах сборки непосредственно во время выполнения.

В результате все тесты выполнялись значительно медленней. Замедление ForwardAll и Static связано с возвращением этими тестами довольно-таки объемного результата. Следовательно, необходимо формирование DataSet'а, копирование данных между процессами и преобразование типов этих данных. Другой неприятностью стало переполнение памяти при попытке вернуть более 200 000 строк. Низкая производительность остальных тестов вызвана большим количеством вызовов методов COM+ объекта. Результаты теста ForwardAll показаны на рис. 2, остальных тестов – в табл. 2.

Если сравнить результаты из табл. 1 и табл. 2, видно, что большинство времени уходит на активизацию и вызов промежуточного объекта.

| | ADO.NET (ODBC) | ADO.NET (MySQL) |
|---|---|---|
| Select100 | 23,187 | 17,483 |
| Static | 28,384/0.009 | 25,184/0.009 |
| Insert 1000 ManyConnection | 23,345 | 18,648 |
| Update 1000 ManyConnection | 23,916 | 18,057 |
| Delete 1000 ManyConnection | 24,165 | 18,246 |

Таблица 2. Быстродействие доступа к данным технологии ADO.NET в локальном COM+ режиме (с)

## Заключение

В работе проведено исследование основных технологий доступа к данным для операционной системы Windows. Учитывая жесткие требования к производительности современных систем, главным критерием оценки было быстродействие. В ходе исследования ODBC проявила себя как одна из самых быстродействующих. Эта технология универсальна и до сих пор применяется не только для операционной системы Windows. Быстродействие следующей технологии, ADO, несколько хуже. Развитием ADO, ориентированным на решение проблем, связанных с разработкой веб-систем и устраняющим многие недостатки устаревшей технологии, стало ADO.NET. Несомненно, пока специалисты из Microsoft не придумают что-то новое, именно эта технология станет главной для доступа к данным для операционной системы Windows.

## Литература


1. Михаилов С. Сравнение скорости доступа к данным (ADO.NET, ADO, ascDB). // RSDN Magazine. 2003. № 3. С. 3–34.
2. Пирогов В. Ю. Ms SQL Server 2000 управление и программирование. СПб: БХВ - Петербург, 2005. 608 с.
3. Арчер Т., Уайтчепел Э. Visual C++.NET. Библия пользователя. М-СПб-К.: Вильямс, 2005. 1216 с.
4. Кэнту М. Delphi 7. Для профессионалов. СПб: Питер, 2003. 840 с.
5. Фролов А.В., Фролов Г.В. Визуальное проектирование приложений C#. М.: Кудиц – образ, 2003. 512 с.
6. Троэлсен Э. C# и платформа.NET. СПб: Питер, 2006. 796 с.

# РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ ОБЩИМ ДОСТУПОМ В ИНТЕРНЕТ

**Д.В. Пудов**
**Научный руководитель – к.т.н., доцент Б.А. Крылов**

В статье рассмотрено применение механизма делегирования посредством разработки системы управления общим доступом в Интернет. Представлен анализ и выбор ОС

## Введение

Использование ресурсов сети Интернет является необходимым условием для развития бизнеса. Для руководства предприятия, одной из основных задач по организации труда является обеспечение доступом в Интернет всех сотрудников. Для решения этой задачи необходимы два компонента – интернет-соединение и сервер общего доступа в Интернет. Постоянное соединение предоставляет интернет-провайдер, а организация общего доступа к нему возлагается на сетевого администратора предприятия.

Организацию общего доступа к сети Интернет обычно можно разделить на два основных этапа – ввод в эксплуатацию и использование. На этапе ввода в эксплуатацию проводятся работы по установке и конфигурированию сервера доступа, поэтому без привлечения к работе сетевого администратора не обойтись. На этапе использования происходит управление политикой доступа к глобальной сети: управление учетными записями пользователей, генерация отчетов об использовании Интернет и др.

С учетом ограниченных бюджетов (особенно малых предприятий) содержание сетевого администратора в штате сотрудников для управления политикой доступа станет причиной излишних финансовых затрат. Имеется более экономичный подход – использование механизма делегирования полномочий. Суть данного подхода заключается в передаче каких-либо функций кому-либо. В применении к данной проблеме делегирование будет выглядеть таким образом: сетевой администратор привлекается на этапе ввода в эксплуатацию сервера доступа, а на этапе использования «делится» полномочиями с ответственным сотрудником предприятия.

Для использования механизма делегирования необходима система управления общим доступом в Интернет, которая позволит сотруднику без глубоких знаний предметной области выполнять возложенные на него функции управления политикой доступа в Интернет.

## Постановка задачи

Необходимо разработать систему управления (СУ) общим доступом в Интернет посредством web-интерфейса администрирования. СУ должна предоставлять безопасный, простой и понятный интерфейс пользователя со следующими функциональными возможностями:

- управление системной службой: запуск, остановка и перезапуск;
- управление правами доступа в Интернет: задание разрешений для пользователей и групп на доступ к www-ресурсам;
- управление ограничениями на доступ в Интернет: задание квот на доступную полосу Интернет-канала;
- просмотр отчетов об использовании Интернет.

На рис. 1 представлена типичная модель сетевого взаимодействия локальной сети предприятия и сети Интернет.

Под безопасным интерфейсом системы управления понимается выполнение двух условий. Во-первых, интерфейс администрирования должен предоставлять минималь-

но необходимый набор функций управления, достаточный для управления политикой доступа в Интернет. Во-вторых, при доступе к СУ необходимо использовать безопасные сетевые протоколы, так как доступ к интерфейсу администрирования может осуществляться из внешней (по отношению к локальной сети предприятия) сети.

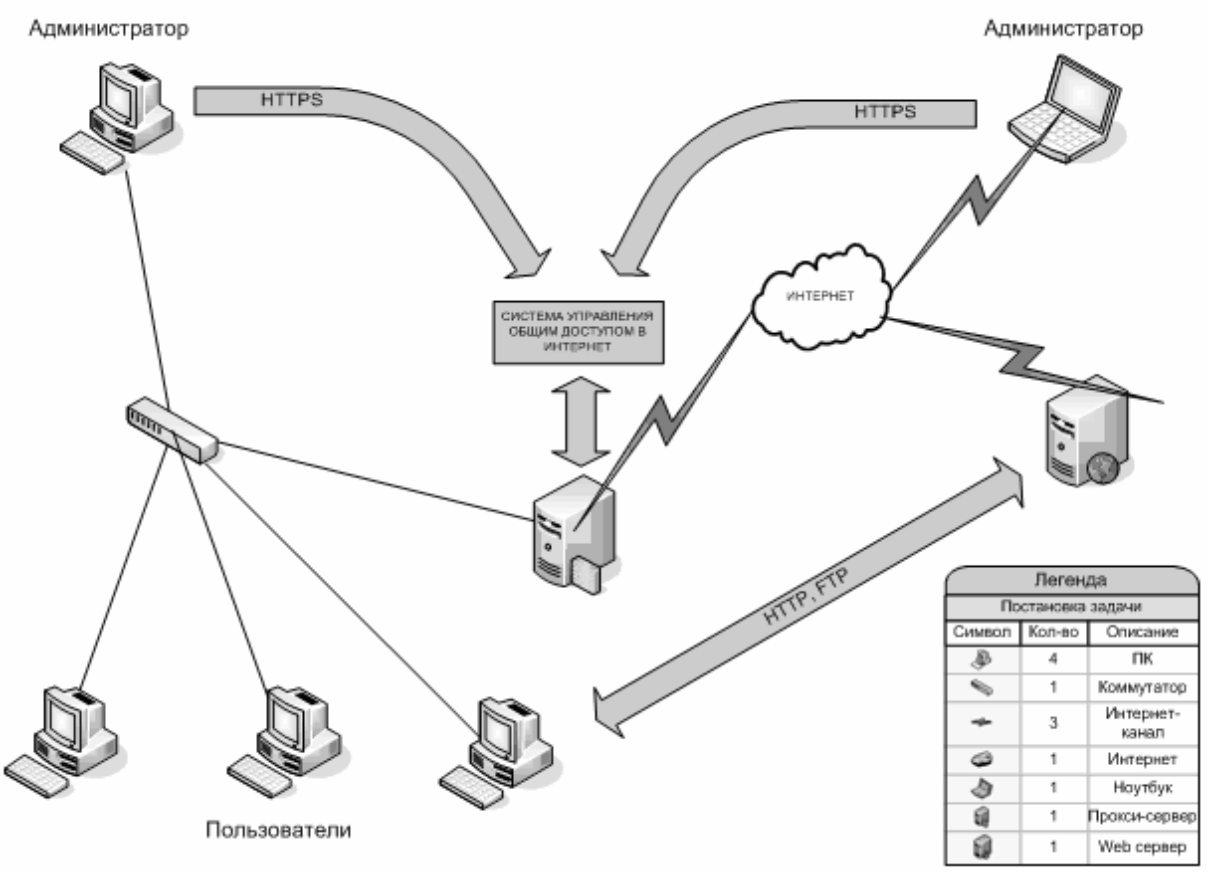



Рис. 1. Постановка задачи

Доступ к системе управления осуществляется по протоколу HTTP, который не является защищенным протоколом – данные передаются в открытом виде. Таким образом, существует возможность перехвата третьей стороной аутентификационных данных, используемых для доступа к системе. Для сохранности передаваемых данных по сетям общего пользования может использоваться протокол SSL (Secure Sockets Layer, уровень безопасных сокетов) [1]. Совместное использование протоколов HTTP и SSL позволяет безопасно использовать сети общего пользования.

## Реализация

Для реализации системы управления общим доступом в Интернет было выбрано следующее программное обеспечение:

- операционная система: Debian GNU/Linux 3.1;
- Web-сервер: Apache 2.0.59;
- прокси-сервер: Squid 2.6;
- язык программирования: Python 2.3.

Взаимодействие указанных компонентов и системы управления представлено на рис. 2.

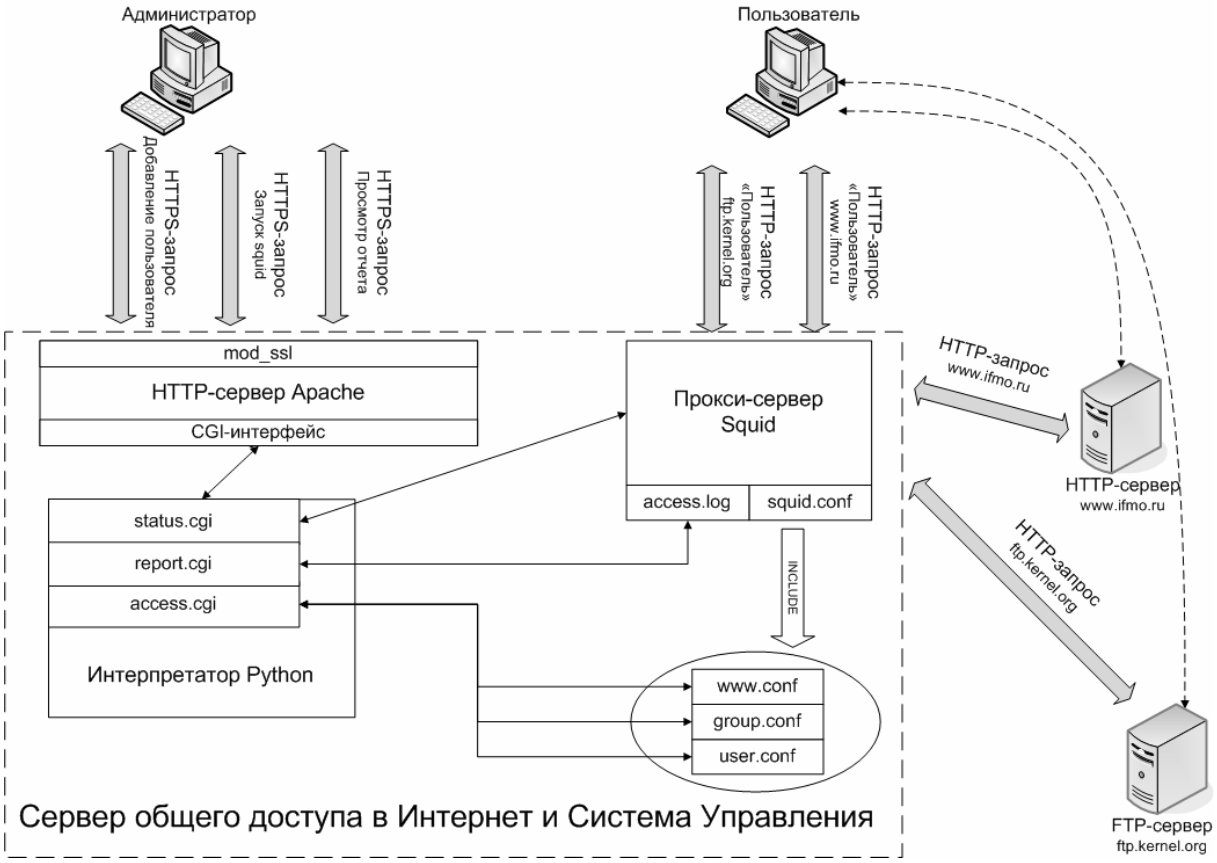



Рис. 2. Структура системы

Система управления реализована как набор CGI-сценариев на Python [1, 2]. В этот набор входит:

- управление системной службой – status.cgi. Выполняет операции остановки или запуска процесса squid;
- управление политикой доступа – access.cgi. Обеспечивает реализацию политики доступа в Интернет;
- просмотр отчетов – report.cgi. Генерация отчетов об использовании Интернет-канала.

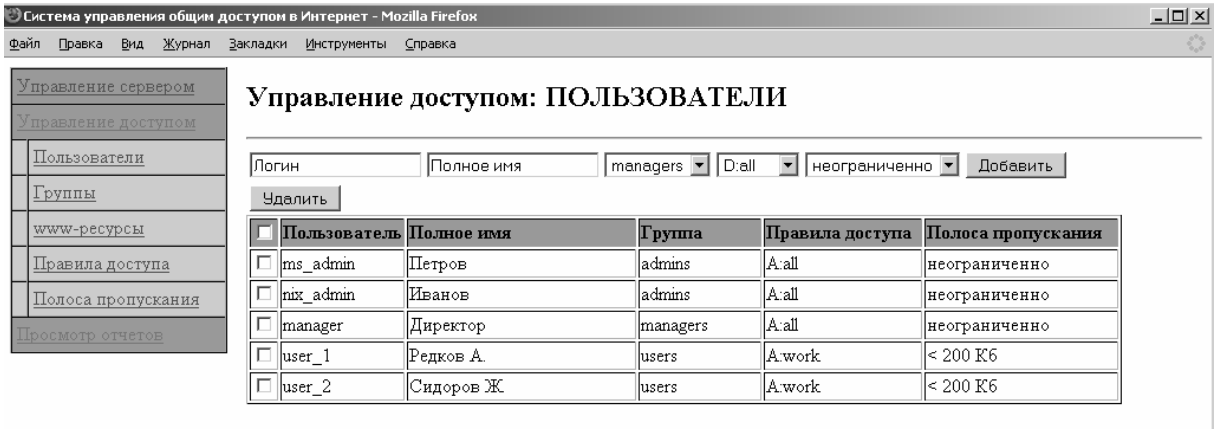



Рис. 3. Web-интерфейс администрирования

Для применения протокола HTTPS был использован модуль web-сервера apache – mod_ssl [3]. Теперь передаваемые данные шифруется по протоколу SSL, таким образом можно безопасно осуществлять доступ к интерфейсу СУ из-за пределов локальной сети предприятия. Для предоставления общего доступа в Интернет используется программ-

ное обеспечение: squid 2.6. Перед делегированием полномочий необходимо выполнить предварительную настройку данного пакета [4].

При разработке интерфейса пользователя были использован HTTP и технологии CGI (Common Gateway Interface, интерфейс общего шлюза) и SSI (Side Server Include, Включения на стороне сервера) для получения динамического интерфейса [5]. Web-интерфейс пользователя представлен на рис. 3.

## Заключение

Проведена разработка и отладка системы управления общим доступом в Интернет. Система имеет простой, интуитивно понятный web-интерфейс. Применение протокола HTPPS предоставляет безопасный доступ к интерфейсу администрирования через сети общего пользования. Данную разработку можно использовать для делегирования полномочий сотруднику без глубоких знаний предметной области, тем самым снижая затраты на поддержание сетевой инфраструктуры.

## Литература


1. Гулич С., Гундаварам Ш., Бирзнекс Г. CGI программирование на Perl. 2-е изд. СПб: Символ-Плюс, 2001. 469 с.
2. Martelli A., Ascher D. Python Cookbook. O'Reilly, 2002. 677 с.
3. Арнольд М. Алмейда Дж. Д., Миллер К. Администрирование Apache. М.: Лори, 2002. 418 с.
4. Wessels D. Squid: The Definitive Guide. O'Reilly, 2004, 464 с.
5. Джамса К. Эффективный самоучитель по креативному Web-дизайну. HTML, XHTML, CSS, JavaScript, PHP, ASP, ActiveX. Текст, графика, звук и анимация. М.: ДиаСофт, 2005. 664 с.

# ТЕСТИРОВАНИЕ ТРАСС СТРУКТУРИРОВАННЫХ КАБЕЛЬНЫХ СИСТЕМ

**Д.А. Шилкин**
**Научный руководитель – Д.О. Изумрудов (ЗАО «Эврика»)**

В статье рассмотрено назначение тестирования современных кабельных систем, а также описаны основные параметры, учитываемые при проверке соответствия их указанной категории и достоверности присущих им характеристик, описанных в международных, европейских или американских стандартах. Дано математическое описание основных функции, необходимое для полного и четкого представления о результатах, на основе которых делается вывод о качестве кабельных трасс.

## Введение

Практически все современные информационные системы и технологии тем или иным образом связаны с необходимостью обмена информацией и использования каналов связи различных типов. Непосредственно каналы для передачи и приема информации, на основе кабельных систем должны иметь строгую иерархическую структуру. На сегодняшний день существуют три нормативных документа и стандарта, описывающих понятия, назначения, принципы построения, а также требования, предъявляемые к кабельным системам, которые получили название структурированные. Любая серьезная и перспективная структурированная кабельная система (СКС) на этапе подготовки к эксплуатации должна быть протестирована специальным оборудованием для выявления тех или иных дефектов и на соответствие заданным характеристикам, определяющихся заранее. Этот завершающий этап построения СКС имеет фундаментальное значение. От его результатов зависит как качество и долговечность установленной системы в целом, так и быстрота и мобильность передачи самой информации между оконечным оборудованием и подключенными пользователями. Основной же практический интерес для тестирования имеют линии связи на основе витых пар и оптики. Они определены в стандарте ISO/IEC 11801 и в бюллетене TSB-67 [1].

В данной работе дано описание параметров кабельных систем, предложенных мировыми стандартами для определения характеристики каналов на разных этапах тестирования. Приведен математический аппарат для более четкого определения каждой функции в системе результирующих данных полученных от устройства тестирования. Даны описания наиболее функциональных и удобных кабельных тестеров для медных и оптических линий.

## Понятие структурированной кабельной системы

Структурированная кабельная система – это универсальная кабельная система здания, группы зданий, предназначенная для использования достаточно длительный период времени без реструктуризации.

Универсальность СКС подразумевает использование ее для различных систем:

1. компьютерная сеть;
2. телефонная сеть;
3. охранная система;
4. пожарная сигнализация.

Такая кабельная система независима от оконечного оборудования, что позволяет создать гибкую коммуникационную инфраструктуру предприятия. Избыточность подразумевает организацию новых рабочих мест без прокладки дополнительных кабельных линий, для чего СКС должна строится с запасом. Гарантированный срок эксплуатации – около 10–15 лет, что включает в себя понятие длительного использования без реструктуризации [1].

Фактически структурированная кабельная система – это совокупность пассивного коммуникационного оборудования:

1. кабель – этот компонент используется как среда передачи данных СКС. Кабель различают на экранированный и неэкранированный;
2. розетки – этот компонент используют как точки входа в кабельную сеть здания;
3. коммутационные панели – используются для администрирования кабельных систем в коммутационных центрах этажей и здания в целом;
4. коммутационные шнуры – используются для подключения офисного оборудования в кабельную сеть здания, организации структуры кабельной системы в центрах коммутации.

Для облегчения проектирования и дальнейшего обслуживания кабельной системы были разработаны международные и европейские стандарты на структурированную кабельную систему (СКС):

1. международный стандарт ISO/IEC 11801 Generic Cabling for Customer Premises;
2. европейский стандарт EN 50173 Information Technology Generic Cabling Systems;
3. американский стандарт ANSI/TIA/EIA 568-A/568-B Commercial Building Telecommunication Cabling Standard.

Стандарты призваны служить общественным интересам, устраняя недопонимание между производителями и потребителями, обеспечивая взаимозаменяемость и универсальное качество продукции, наряду с ее доступностью и грамотным использованием. Стандарты телекоммуникационной инфраструктуры зданий должны обеспечить работу разнотипного оборудования любых производителей, создание кабельных системы на этапе строительства зданий и их длительную эксплуатацию.

## Категории и классы кабельной системы

При классификации кабельных систем по производительности существуют некоторые отличия, связанные с различным подходами к этому вопросу в международном ISO 11801 и американском стандарте TIA/EIA-568-A. Международный стандарт определяет классы приложений (от A до F), которые могут функционировать по данной системе, а американский стандарт специфицирует системы по максимальной частоте передаваемых сигналов (категория 3–6). Поэтому для определения кабельных систем используются как категории, так и классы.

| Частота передачи сигналов, Мгц | ISO 11801, приложения | TIA/EIA-568-A, Категории |
|---|---|---|
| 0,1 | класса A | |
| 1 | класса B | |
| 16 | класса C | 3 |
| 20 | | 4 |
| 100 | класса D | 5 |
| 100 | класса D+ | 5e |
| 250 | класса E | 6 |
| 600 | класса F | 7 |

Таблица 1. Категории и классы кабельных систем

Стандарты определяют среду передачи, параметры разъемов, линии и канала, в том числе предельно допустимые длины, способы подключения проводников (последовательность), топологию и функциональные элементы СКС. В настоящее время стандарты на категорию 6 и 7 находятся в стадии доработки и утверждения, поэтому существует вероятность изменения характеристик для этих систем.

## Тестирование каналов и стационарных линий СКС

Комплекс измерений параметров отдельных электрических и оптических компонентов СКС, а также смонтированных линий на их основе предназначен для определения состояния СКС, предупреждения повреждений, накопления статистических данных, используемых при разработке мероприятий по повышению надежности связи и, наконец, является обязательной процедурой перед вводом СКС в эксплуатацию. Измерения производятся в следующих случаях:

1. в процессе выполнения входного контроля отдельных компонентов перед началом работ по их монтажу;
2. при проведении приемо-сдаточных испытаний СКС;
3. во время эксплуатации кабельной системы при выполнении профилактических, аварийных и контрольных проверок.


Рис. 1. Структура видов измерений в СКС

Передаточные характеристики трактов, которые создаются с помощью СКС для передачи сигналов различных инженерных систем, обсуждаются и оцениваются в стандарте ISO/IEC 11801:2002(E) с помощью понятий *канал* (англ. channel) и *стационарная линия* (англ. permanent link). Под каналом понимается тракт передачи сигналов по СКС от одного активного оборудования до другого. Сами вилки на концах кабелей не входят в состав канала, поскольку их влияние на систему в целом учтено разработчиками активного оборудования [2]. Под стационарной линией понимается часть тракта передачи сигналов по СКС, включающая в себя лишь стационарный кабель и соединители на его концах [2].

### Основные параметры тестирования линий кабельных систем

Затухание (Attenuation) – потеря мощности сигнала. Это отношение мощности сигнала на выходе передатчика к мощности сигнала на входе приемника, выраженное в децибелах (дБ). Чем меньше затухание, тем сильнее сигнал на входе приемника, тем лучше связь.

$A = \lg (P_1 / P_2)$,

где $P_1$ – мощность сигнала на выходе передатчика, $P_2$ – мощность сигнала на входе приемника.

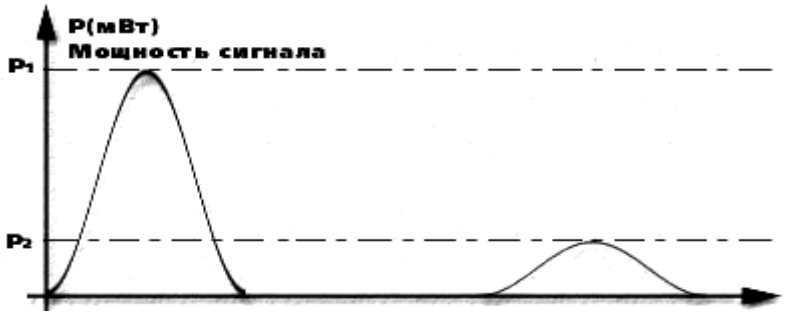



Рис. 2. Распределение мощности сигнала

$A = 10 \lg (P_1 / P_2) = 10 \lg (U_1 I_1 / U_2 I_2) = 10 \lg (U_1 * (U_1 / R_1) / U_2 * (U_2 / R_2))$,

принимаем $R_1 = R_2$, таким образом, $A = 10 \lg (U_1^2 / U_2^2) = 20 \lg (U_1 / U_2)$.

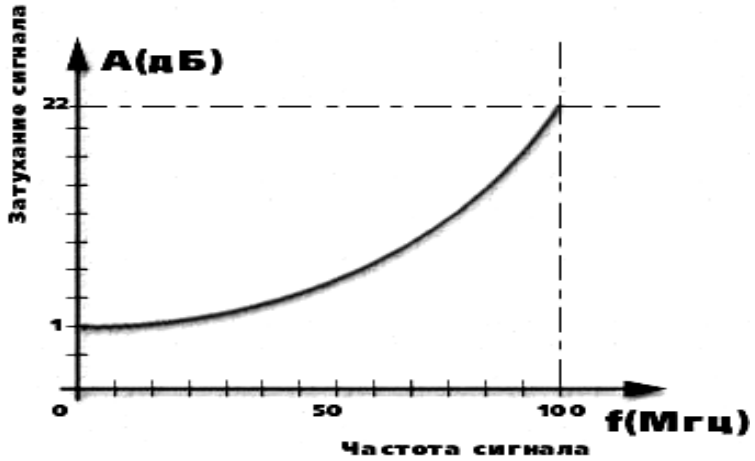



Рис. 3. Зависимость затухания от частоты сигнала

Тестирование производится для всего диапазона рабочих частот. Оценка результата тестирования для каждой пары выводится на основании наихудшего результата. Затухание канала и базовой линии является суммой затуханий, вносимых всеми их составляющими элементами: горизонтальным кабелем, оконечными и коммутационными шнурами и разъемами. Максимально допустимое затухание можно выразить следующим образом:

$$A = \Sigma A_{\text{разъема}} + A_{\text{кабеля на 100м}} * (L_{\text{кабеля}} + 1{,}2 * \Sigma L_{\text{шнуров}}) / 100,$$

где $\Sigma A_{\text{разъема}}$ – сумма максимально допустимых затуханий, вносимых всеми разъемами (в канале может быть до четырех разъемов, в базовой линии всегда два разъема); $A_{\text{кабеля на 100м}}$ – максимально допустимое затухание горизонтального кабеля на длине 100 м; $L_{\text{кабеля}}$ – фактическая длина горизонтального кабеля канала или базовой линии; $\Sigma L_{\text{шнуров}}$ – фактическая сумма длин всех шнуров канала или базовой линии [1].

| Частота, МГц | Максимально допускаемые значения A, дБ | | | | | |
|---|---|---|---|---|---|---|
| | Класс A | Класс B | Класс C | Класс D | Класс E | Класс F |
| 0.1 | 16,0/16,0 | 5,5/5,5 | – | – | – | – |
| 1 | – | 5,8/5,8 | 4,2/4,2 | 4,0/4,0 | 4,0/4,0 | 4,0/4,0 |
| 16 | – | – | 14,4/12,2 | 9,1/7,7 | 8,3/7,1 | 8,1/6,9 |
| 100 | – | – | – | 24,0/20,4 | 21,7/18,5 | 20,8/17,7 |
| 250 | – | – | – | – | 35,9/30,7 | 33,8/28,8 |
| 600 | – | – | – | – | – | 54,6/46,6 |

Таблица 2. Максимально допустимые стандартом ISO/IEC 11801:2002(E) значения A, дБ, в формате канал/стационарная линия

При прохождении сигнала по витой паре создается электромагнитное поле, которое взаимодействует с сигналами, передаваемыми по соседним парам. В зависимости от того, осуществляется двунаправленная или однонаправленная передача сигнала, важно оценить влияние помех, наведенных сигналом на ближнем или на дальнем конце, по отношению к источнику сигнала [3]. Ослабление перекрестных наводок на ближнем от передатчика конце, *NEXT* (Near End Crosstalk loss) – это параметр двунаправленной передачи, характеризующий затухание сигнала помехи, наведенного сигналом передатчика на смежную пару. Измеряется в децибелах (дБ). Чем выше значение *NEXT*, тем меньше влияние помех между двумя парами проводников. Минимально допустимое значение NEXT для канала рассчитывается по формуле:

$$NEXT = -20\lg \cdot (10^{-NEXT\text{кабеля} / 20} + n \cdot 10^{-NEXT\text{разъема} / 20}),$$

где $NEXT_{\text{кабеля}}$ – минимальное переходное затухание горизонтального кабеля на длине 100 м; $NEXT_{\text{разъема}}$ – минимальное переходное затухание разъема; $n$ – количество разъемов на ближнем конце.


Рис. 4. Однонаправленная передача сигнала

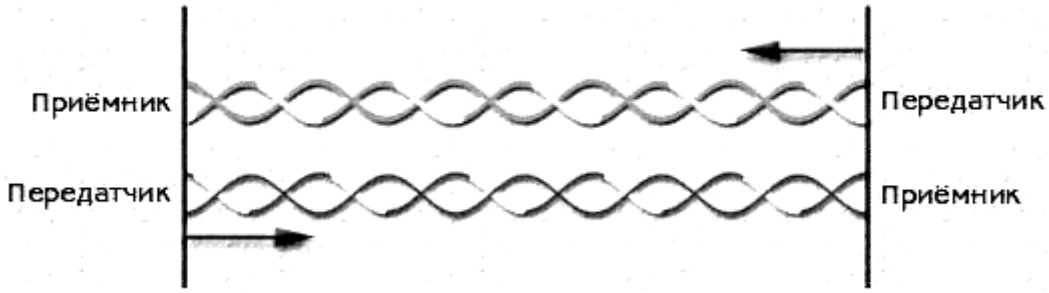



Рис. 5. Двунаправленная передача сигнала

Параметр *NEXT* специфицирован для всех классов и должен измеряться на обоих концах тракта. Он также не зависит от длины кабеля, а определяется только конкретным взаимным влиянием пар [2].


Рис. 6. Наводки, определяемые параметром NEXT

| Частота, МГц | Максимально допускаемые значения NEXT, дБ | | | | | |
|---|---|---|---|---|---|---|
| | Класс A | Класс B | Класс C | Класс D | Класс E | Класс F |
| 0.1 | 27,0/27,0 | 40,0/40,0 | – | – | – | – |
| 1 | – | 25,0/25,0 | 39,1/39,1 | 60,0/60,0 | 65,0/65,0 | 65,0/65,0 |
| 16 | – | – | 19,4/21,1 | 43,6/45,2 | 53,2/54,6 | 65,0/65,0 |
| 100 | – | – | – | 30,1/32,3 | 39,9/41,8 | 62,9/65,0 |
| 250 | – | – | – | – | 33,1/35,3 | 56,9/60,4 |
| 600 | – | – | – | – | – | 51,2/54,7 |

Таблица 3. Максимально допустимые стандартом ISO/IEC 11801:2002(E) значения NEXT, дБ, в формате канал/стационарная линия

Современные высокоскоростные приложения используют одновременную передачу и прием информации по всем четырем парам. Помимо параметра *NEXT* , в этом

случае необходимо учитывать влияние помех на дальнем от передатчика конце линии. Ослабление перекрестных наводок на дальнем от передатчика конце, *FEXT* (Far End Crosstalk loss) – это параметр однонаправленной передачи, характеризующий затухание сигнала помехи, наведенного сигналом передатчика на смежную пару. Измеряется в децибелах (дБ). Чем выше значения *FEXT*, тем меньший уровень имеет наводка в соседних парах и тем лучше качество передачи. Тестирование проводится для двух концов кабельной цепи.

$FEXT = 20\lg \cdot (U_{FEXT}/U_0)$,

где $U_{FEXT}$ – амплитуда импульса наведенного на конце пары; $U_0$ – амплитуда импульса, передаваемого по паре.

Параметр *FEXT,* как правило, сам по себе не измеряется на инсталлированной СКС, но используется для определения других параметров тракта на дальнем конце [2].


Рис. 7. Наводки, определяемые параметром FEXT

Для передачи информации в высокоскоростных приложениях используются одновременно несколько пар. Сигналы нескольких приложений все чаще передаются в одном многопарном кабеле. На приемник одновременно действует несколько источников помех. Поэтому вводится еще один параметр, так называемая суммарная мощность. Первый из них – это потери *NEXT* суммарной мощности *PS-NEXT* (Power sum *NEXT*). Параметр важен при двунаправленной передаче. Это величина наведенного на одну из пар от нескольких передатчиков нежелательного помехового сигнала, измеренного на ближнем от передатчиков конце. Тестирование *NEXT* суммарной мощности дает один результат на каждую пару кабеля. Результаты потерь *NEXT* суммарной мощности необходимо определять для обоих концов кабеля, так как передачи идут с обоих концов кабельной системы:

$$PSNEXT\,(k) = -20\lg \cdot \sum_{i=1, i\neq k}^{n} 10^{-0{,}1NEXT\,(i,\,k)},$$

где $n$ – число пар; $PSNEXT\,(k)$ – «суммарный» параметр «возмущаемой» пары $k$; $NEXT\,(i, k)$ – параметр *NEXT* для «возмущаемой» пары $k$ и «возмущающей» пары $i$.

Параметр *PSNEXT* вычисляется (но не измеряется непосредственно) по измеренным значениям параметра *NEXT* [2].

Следующий параметр – это потери *FEXT* суммарной мощности *PS-FEXT* (Power sum *FEXT*). Параметр важен при однонаправленной передаче. Это величина наведенного на одну из пар от нескольких передатчиков нежелательного помехового сигнала, измеренного на дальнем от передатчиков конце. Тестирование *FEXT* суммарной мощности дает один результат на каждую пару кабеля.

$$PSFEXT\,(k) = -20\lg \cdot \sum_{i=1, i\neq k}^{n} 10^{-0{,}1FEXT\,(i,\,k)},$$

где $n$ – число пар; $PSFEXT\,(k)$ – «суммарный» параметр «возмущаемой» пары $k$; $FEXT\,(i, k)$ – параметр *NEXT* для «возмущаемой» пары $k$ и «возмущающей» пары $i$.

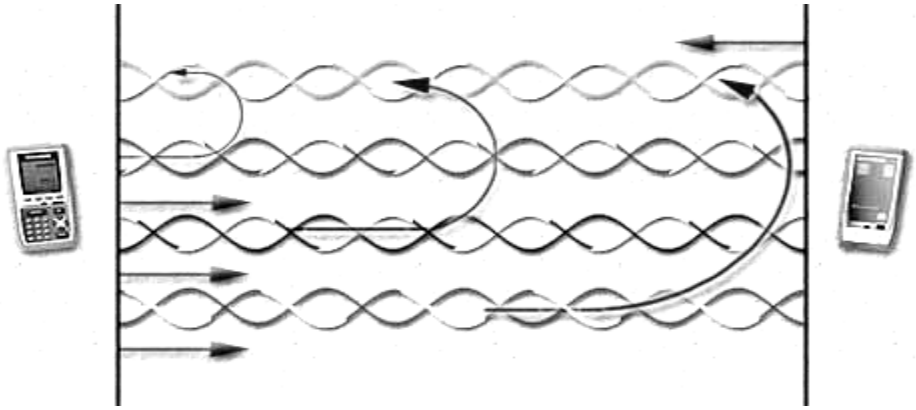


Рис. 8. Наводки, определяемые параметром PSNEXT

| Частота, МГц | Максимально допускаемые значения PSNEXT, дБ | | |
|---|---|---|---|
| | Класс D | Класс E | Класс F |
| 1 | 57,0/57,0 | 62,0/62,0 | 62,0/62,0 |
| 16 | 40,6/42,2 | 50,6/52,2 | 62,0/62,0 |
| 100 | 27,1/29,3 | 37,1/39,3 | 59,9/62,0 |
| 250 | – | 30,2/32,7 | 53,9/57,4 |
| 600 | – | – | 48,2/51,7 |

Таблица 4. Максимально допустимые стандартом ISO/IEC 11801:2002(E) значения PSNEXT, дБ, в формате канал/стационарная линия

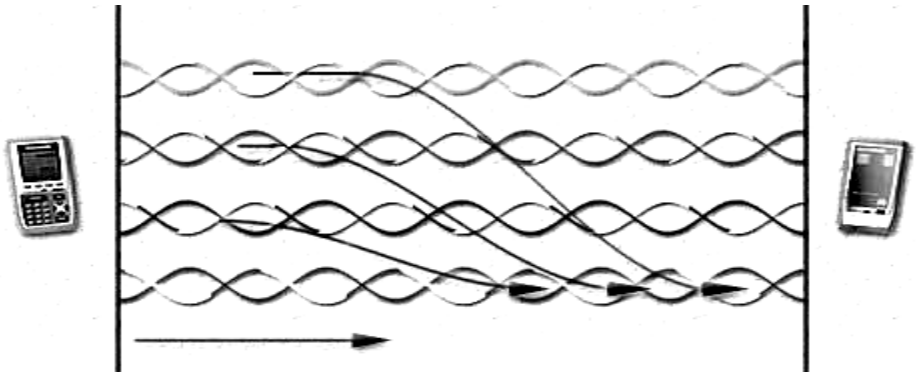


Рис. 9. Наводки, определяемые параметром PSFEXT

Существуют также параметры, которые являются опциональными при тестировании кабельных трактов. Параметр *FEXT*, в отличие от *NEXT*, зависит от длины линии на всем ее протяжении. Две линии с использованием одних и тех же элементов, но разной длины, будут иметь разные значения *FEXT*. Поэтому нормируется параметр – равноуровневые перекрестные наводки на дальнем конце или отношение затухания к суммарным двунаправленным наводкам – *ELFEXT* (Equal-level Far End Crosstalk). Этот параметр выражается в децибелах (дБ) [3]. На экран сканера выводятся результаты, рассчитанные как разность между измеренными потерями *FEXT* и затуханием сигнала в возмущаемой паре:

$$ELFEXT\,(i,\,k) = FEXT\,(i,\,k) - A\,(k),$$

где $i$ – номер «возмущающей» пары; $k$ – номер «возмущаемой» пары; $A\,(k)$ – затухание «возмущаемой» пары.

| Частота, МГц | Максимально допускаемые значения ELFEXT, дБ | | |
|---|---|---|---|
| | Класс D | Класс E | Класс F |
| 1 | 57,4/58,6 | 63,3/64,2 | 65,0/65,0 |
| 16 | 33,3/34,5 | 39,2/40,1 | 57,5/59,3 |
| 100 | 17,4/18,6 | 23,3/24,2 | 44,4/46,0 |
| 250 | – | 15,3/16,2 | 37,8/39,2 |
| 600 | – | – | 31,3/32,6 |

Таблица 5. Максимально допустимые стандартом ISO/IEC 11801:2002(E) значения ELFEXT, дБ, в формате канал/стационарная линия

Потери суммарной мощности или же отношение затухания к суммарным однонаправленным наводкам – *PSELFEXT* (Power sum *ELFEXT*). Характеризует превышение уровня сигнала над уровнем однонаправленных наводок от всех пар. Равноуровневые перекрестные помехи на дальнем конце модели суммарной мощности – величина наведенного на одну из пар от нескольких передатчиков, нежелательного помехового сигнала, измеренного на дальнем от передатчиков конце относительно уровня принимаемого сигнала, измеренного на той же паре. Тестирование *PSELFEXT* суммарной мощности дает один результат на каждую пару кабеля. Результаты потерь *PSELFEXT* необходимо определять для обоих концов кабеля, т. к. передачи идут с обоих концов кабельной системы [3].

$$PSELFEXT\,(k) = -10\lg \cdot \sum_{i=1, i\neq k}^{n} 10^{-ELFEXT\,(i,\,k)\,/\,10},$$

где $i$ – номер «возмущающей» пары; $k$ – номер «возмущаемой» пары; $ELFEXT$ ($i$, $k$) – параметр, обусловленный воздействием пары $i$ на пару $k$.

| Частота, МГц | Максимально допускаемые значения PSELFEXT, дБ | | |
|---|---|---|---|
| | Класс D | Класс E | Класс F |
| 1 | 54,4/55,6 | 60,3/61,2 | 62,0/62,0 |
| 16 | 30,3/31,5 | 36,2/37,1 | 54,5/56,3 |
| 100 | 14,4/15,6 | 20,3/21,2 | 41,4/43,0 |
| 250 | – | 12,3/13,3 | 34,8/36,2 |
| 600 | – | – | 28,3/29,6 |

Таблица 6. Максимально допустимые стандартом ISO/IEC 11801:2002(E) значения PSELFEXT, дБ, в формате канал/стационарная линия

Соотношение затухания и переходного затухания на ближнем конце характеризует отношение «сигнал/помеха» на приемном конце линии. Другими словами *ACR* (Attenuation to Crosstalk Ratio) означает превышение сигнала над уровнем собственных шумов для двунаправленной передачи сигналов (для сравнения *ELFEXT* – для однонаправленной). Отношение затухания к наводкам *ACR* в логарифмическом виде – это разность *NEXT* и затуханием сигнала линии [2].

$$ACR\,(i, k) = NEXT\,(i, k) - A\,(k),$$

где $i$ – номер «возмущающей» пары; $k$ – номер «возмущаемой» пары; $NEXT$ ($i$, $k$) – параметр *NEXT*, обусловленный воздействием пары $i$ на пару $k$; $A$ ($k$) – параметр $A$ для пары $k$.

| Частота, МГц | Максимально допускаемые значения ACR, дБ | | |
|---|---|---|---|
| | Класс D | Класс E | Класс F |
| 1 | 56,0/56,0 | 61,0/61,0 | 61,0/61,0 |
| 16 | 34,5/37,5 | 44,9/47,5 | 56,9/58,1 |
| 100 | 6,1/11,9 | 18,2/23,3 | 42,1/47,3 |
| 250 | – | –2,8/4,7 | 23,1/31,6 |
| 600 | – | – | –3,4/8,1 |

Таблица 7. Максимально допустимые стандартом ISO/IEC 11801:2002(E) значения ACR, дБ, в формате канал/стационарная линия

Это важнейший параметр передачи сигналов. В типичном случае (протоколы 10 BaseT Ethernet, 100 BaseT Fast Ethernet и другие) используются две пары. Передача сигналов идет в противоположных направлениях. Отсюда перекрестные или двунаправленные наводки.

Следующий параметр – отношение затухания к суммарным двунаправленным наводкам. Потери *ACR* (Attenuation to Crosstalk Ratio) суммарной мощности *PS–ACR* (Power sum *ACR*). Параметр, характеризующий превышение уровня сигнала над уровнем двунаправленных наводок от всех пар.

$PSACR\ (k) = \mathrm{PS}NEXT\ (k) - A\ (k)$,

где $k$ – номер «возмущаемой» пары; $PSNEXT\ (k)$ – «суммарный» параметр пары $k$; $A\ (k)$ – затухание пары $k$.

Равноуровневые перекрестные помехи на дальнем конце модели суммарной мощности – величина наведенного от нескольких передатчиков, нежелательного помехового сигнала, измеренного на ближнем от передатчиков конце, на одну из пар относительно уровня принимаемого сигнала, измеренного на той же паре. Тестирование *PS-ACR* дает один результат на каждую пару кабеля [3].

| Частота, МГц | Максимально допускаемые значения PSACR, дБ | | |
|---|---|---|---|
| | Класс D | Класс E | Класс F |
| 1 | 53,0/53,0 | 58,0/58,0 | 58,0/58,0 |
| 16 | 31,5/34,5 | 42,3/45,1 | 53,9/55,1 |
| 100 | 3,1/8,9 | 15,4/20,8 | 39,1/44,3 |
| 250 | – | –5,8/2,0 | 20,1/28,6 |
| 600 | – | – | –6,4/5,1 |

Таблица 8. Максимально допустимые стандартом ISO/IEC 11801:2002(E) значения PSACR, дБ, в формате канал/стационарная линия

Реальная кабельная линия всегда имеет неоднородности, которые приводят к отражению электромагнитной волны, в процессе прохождения сигнала по кабелю. Обратные потери (Return Loss) – мера величины отражения сигналов, вызываемого несоответствием импедансом компонентов кабельной системы. Этот параметр определяется как отношение мощности основного сигнала к мощности обратного потока энергии. Измеряется в децибелах. Величина обратных потерь имеет особенно важное значение для работы приложений, использующих технологии синхронной двухсторонней передачи сигналов.

$RL = 20\lg \cdot (|U_{RL}| / U_0)$,

где $|U_{RL}|$ – амплитуда отраженного импульса (по модулю); $U_0$ – амплитуда входного импульса линии.

Стоит заметить, что поскольку для амплитуд $U_{RL} < U_0$, то значения логарифма всегда отрицательны, что необходимо учитывать при расчете величин $U_{RL}$ и $U_0$ [2].

| Частота, МГц | Максимально допускаемые значения RL, дБ | | | |
|---|---|---|---|---|
| | Класс C | Класс D | Класс E | Класс F |
| 1 | 15,0/15,0 | 17,0/19,0 | 19,0/21,0 | 19,0/21,0 |
| 16 | 15,0/15,0 | 17,0/19,0 | 18,0/20,0 | 18,0/20,0 |
| 100 | – | 10,0/12,0 | 12,0/14,0 | 12,0/14,0 |
| 250 | – | – | 8,0/10,0 | 8,0/10,0 |
| 600 | – | – | – | 8,0/10,0 |

Таблица 9. Минимально допустимые стандартом ISO/IEC 11801:2002(E) значения RL, дБ, в формате канал/стационарная линия

Наиболее распространенной причиной возникновения обратных потерь является различие волнового сопротивления у компонентов кабельного канала (розетка, патч-панель, кабель и т.д.) Поэтому рекомендуется подбирать оборудование одного производителя, обладающее одинаковыми (специально подобранными) характеристиками. Также неоднородность может возникнуть в случае нарушения шага скрутки. Это может быть следствием брака при производстве либо ошибки монтажников при протяжке кабеля, надлома жилы или слишком сильного изгиба.

**Принципы тестирования кабельных систем**

Конечные пользователи и проектировщики сетей постоянно планируют более высокие скорости передачи данных, возможности передачи большего количества данных, а также способность сети к гибкой и удобной переконфигурации.

Бытует мнение, что около 20 % высокоскоростных сетей не обеспечивают возможного быстродействия, что является результатом некачественной реализации кабельных систем. Это особенно хорошо заметно на примерах высокоскоростных систем, в состав которых входят Fast Ethernet, коммутируемые LAN, Gigabit Ethernet. Однако некоторые низкоскоростные системы (Ethernet, Token Ring) могут приемлемо функционировать даже при неграмотной инсталляции [4].

**Тестирование сетей на основе медного кабеля**

Измерительное и тестирующее оборудование СКС на основе витых пар можно подразделить на три основные группы:

1. сетевые анализаторы (Network Analyzers);
2. тестеры СКС (FTE-Field Test Analyzers);
3. электрические тестеры или мультиметры (Continuity or Cable Testers).

Сетевые анализаторы представляют собой эталонное измерительное оборудование лля диагностики и сертификации кабелей и кабельных систем. Это прецизионные крупногабаритные и дорогие (стоимостью более 20 тысяч долларов) приборы, предназначенные для использования в лабораторных условиях.

Тестеры СКС были разработаны специально для диагностики и тестирования СКС непосредственно на объекте монтажа кабельной системы (иначе, для выполнения так называемого полевого тестирования – field testing). Достаточно часто их называют кабельными сканерами (Cable Scanners). Они являются основным инструментом для оперативных измерений подсистем СКС, реализованных на основе витых пар. Эти устройства позволяют проводить комплексную проверку 4-парных кабелей, линий классов C, D, E и категорий 3, 4, 5, 6.

Электрические тестеры, или мультиметры, представляют собой простые, дешевые и широко распространенные приборы; позволяют измерять постоянные и переменные ток и напряжение, а также активное сопротивление постоянному току. Наиболее совершенные устройства данной группы дополнительно контролируют частоту, емкость, температуру полевых и биполярных транзисторов.

На сегодняшний день одна из наиболее удобных и важных функций тестирования кабельных систем – автоматизация процесса проведения измерений и интерпретация полученных результатов. Во время общего теста (режим Autotest) в течение нескольких секунд последовательно, без вмешательства оператора, измеряется ряд необходимых для проверки параметров. Результаты измерений сравниваются с требованиями стандартов или определенного сетевого протокола при его указании в явном виде, затем выдается отчет с общим выводом по результатам тестирования в виде ДА/НЕТ (Pass/Fail). Время автотеста составляет примерно 10–20 с. Многие из представленных на рынке сканеров позволяют поддерживать голосовую связь во время тестирования, что достаточно удобно при тестировании кабельной системы, распределенной по большой площади. От устройств этого типа в систему поступает сигнал, который затем сравнивается процессором сканера с пришедшим отраженным сигналом. Достаточно часто для исследования кабельной системы используется подача контрольного сигнала с определенными параметрами, благодаря чему можно достаточно точно определить характер неисправности [4].

**Тестирование электрической подсистемы СКС**

Принцип действия рефлектометра во временной области TDR (Time Domain Reflectometer) основан на анализе сигнала, отраженного от различных неоднородностей в линии при ее зондировании мощными импульсами тока небольшой длительности.

Электрическая волна, возбуждаемая в тестируемой линии импульсным генератором рефлектометра, при распространении в линии отражается в обратном направлении от всех точек неоднородностей. Анализатор приемника контролирует как момент прихода отраженного сигнала, так и изменение его формы во времени. Результат работы анализатора может быть представлен на дисплее графически, в виде так называемой рефлектограммы, или же в табличной форме. По времени задержки между зондирующим и приходящим импульсом рассчитывается расстояние до неоднородности, и его значения выводятся на экран.

Рефлектограммы для электрических кабелей получили достаточно широкое распространение в сетях городской и междугородной связи. Из-за трудностей анализа начального участка они эффективны только в процессе тестирования кабелей магистральных подсистем и поэтому не получили широкого распространения в технике СКС. При тестировании кабельных систем здания их роль успешно выполняют кабельные сканеры, реализующие функции рефлектометра [1].

**Тестирование сетей на основе оптоволоконного кабеля**

Измерительное и тестирующее оборудование СКС на основе оптоволоконных линий можно подразделить на три основные группы:

1. оптические тестеры;
2. рефлектометры;
3. визуальные локаторы.

Оптические тестеры, или измерители оптических потерь, предназначены для измерения среднего уровня мощности оптического излучения на рабочих длинных волн волоконно-оптических линий связи (850, 1300, 1550 нм) и определения затухания сигнала в кабелях и отдельных компонентах линии. Тестеры могут работать как с многомодовыми, так и с одномодовыми световодами и комплектуются одним или несколькими сменными адаптерами для подключения к вилкам разъемов различных типов.

В состав оптического тестера входят два основных прибора: измеритель оптической мощности и источник излучения.

Измерители оптической мощности (optical power meter, OPM) применяются для определения мощности оптического сигнала и затухания сигнала в линиях и каналах. В

составе конструкции измерителя имеются германиевый фотодиод с усилителем фототока, сигнальный процессор и цифровой дисплей. Фотодиод преобразует падающий на его окно световой поток в электрический ток, который обрабатывается сигнальным процессором. Результат обработки выводится на цифровой индикатор.

Стабилизированные источники излучения (Stabilized Light Source, SLS) служат для подачи в волоконно-оптический элемент оптического сигнала заданной мощности и длины волны. Постоянство выходной мощности такого источника поддерживается путем регулировки прямого тока излучателя. Часто применяются отдельные модели оптических источников и измерителей, рассчитанных на одну рабочую длину волны, для уменьшения финансовых затрат пользователей.

Оптические рефлектометры во временной области (Optical Time Domain Reflectometer, OTDR), или просто рефлектометры, являются одним из наиболее мощных средств для тестирования волоконно-оптических линий. В процессе измерения контролируемое волокно зондируют через разветвитель мощными оптическими импульсами небольшой длительности. Из-за отражений от неоднородностей возникает поток обратного рассеяния, которое интерпретируется как затухание через функцию длины [1]. Анализ данных позволяет определить местонахождение неоднородности и величины потерь. Полученные результаты представляются в форме диаграммы (рефлектограммы). Выделим особенности оптических рефлектометров во временной области:

- позволяют за один цикл измерений определять целый ряд параметров – длину, затухание, неоднородности;
- допускают выполнение измерений с одного конца кабеля;
- высокие требования к качеству ввода излучения в тестируемое волокно;
- достаточно медленное время получения рефлектограммы ~30 с;
- высокая стоимость.

Оптический локатор – это упрощенный вариант рефлектометра. Принцип его действия идентичен, а упрощение достигнуто главным образом за счет отказа от графического дисплея и применения более простого программного обеспечения.

Визуализатор дефектов – одна из наиболее полезных функций локаторов и дефектоскопов. Он предназначен для выявления близких к концу кабеля (не более 5 км) обрывов и других дефектов волоконных световодов методом просветки. Основой прибора является лазер красного свечения. При подключении визуализатора к волокну в месте повреждения наблюдается красное свечение.

### Современные устройства для тестирования кабельных систем

Современные устройства для тестирования кабельных систем делятся на уровни точности, которые характеризуют количество параметров, верифицируемых тестером для исследования кабеля на принадлежность определенной категории. Соответствие необходимых уровней точности для устройств тестирования на предмет исследования определенных категорий кабельных систем представлено в табл. 10.

| | |
|---|---|
| Категория 5 (Class D) | II уровень |
| Категория 5e (Class D+) | II-e уровень |
| Категория 6 (Class E) | III уровень |
| Категория 7 (Class F) | VI уровень |

Таблица 10. Соответствие кабельной категории уровню тестера

**Fluke Networks Corporation** выпустила новую серию кабельных анализаторов DTX CableAnalyzer – это платформа для тестирования, которая поддерживает не только нынешние стандарты, но пригодится для анализа сетей в будущем.

Применение новейших разработок и решений для каждого этапа комплексного тестирования значительно сократило время, необходимое для сертификации сети и проверки ее соответствия установленным стандартам. Приборы данной серии могут тестировать как системы на основе медного кабеля, так и волоконно-оптические системы.

Серия кабельных тестеров DTX гарантирует очень высокий уровень точности (Level IV), измерение параметров кабеля осуществляется в частотном диапазоне до 900 МГц, время работы от аккумуляторов – до 12 часов. Большой запас частоты обеспечивает возможность работы с сетями категории 7 (Class F). Простой и интуитивно понятный интерфейс прибора обеспечит очень быстрое конфигурирование, а с помощью поставляемого в комплекте ПО LinkWare достаточно просто создать отчет о результатах сертификации соединений. Проведение процедуры «Автотест» для категории 6 (Class E) составляет 12 секунд (для приборов DTX-1800, DTX-1200) и 30 секунд для DTX-LT. Прибор указывает местонахождение неисправности на любом расстоянии от тестера и подсказывает необходимые действия.

Опционально поставляемые оптические модули (DTX Fiber Module) хорошо защищены внутри корпуса, время выполнения процедуры «Автотест» для оптических соединений составляет 12 секунд. Поддерживается работа как с одномодовыми, так и с многомодовыми линиями связи, сертификация по классу I (TSB140) одновременно двух волоконно-оптических кабелей на двух длинах волн (850 нм и 1300 нм). Внутренняя память прибора позволяет сохранить до 250 отчетов с графиками или до 2000 отчетов в текстовом варианте, а карта расширения памяти на 16 МБ позволит сохранить еще 300 дополнительных отчетов с графиками. Благодаря встроенному USB-порту легко передать данные на компьютер. В комплектацию входит переговорное устройство, позволяющее держать связь при выполнении работ как по медному, так и по оптическому кабелю. Стоимость устройств этой серии составляет от $12000–$18000 [4].


Рис. 10. Тестер Fluke Networks DTX-1800

**Продукты Ideal Industries.** Семейство устройств LANTEK 7G обладает возможностью работы в динамическом диапазоне частот вплоть до 1 ГГц, что позволяет тестировать сети, удовлетворяющие требованиям категорий 6, 6a, 7(Class F) и уровню точности IV. Возможно сохранение до 6000 результатов тестов. Благодаря вспомогательному оборудованию FIBERTEK Accessory и TRACETEK Accessory (технология OTDR) можно тестировать волоконно-оптические системы (одномодовые и многомодовые) на наличие неисправностей и определения их местоположения. Комплект поставки включает в себя PCMCIA-адаптер для установки компактной перепрограммируемой памяти, ко-

торая предоставляет неограниченную возможность для сохранения полученных результатов тестирования, а также для установки дополнительных опций, карту расширения памяти на 64 МБ, USB Port Flash Card Reader. Полезно наличие расширенного пользовательского интерфейса, включая контекстно-зависимую справку. Стоимость этого набора составляет в базовой комплектации $7000 и в Premium-комплектации – $8000 [4].


Рис. 11. Тестер Ideal Industries Lantek 7G

Продукт Acterna. LT 8600 обеспечивает уровень точности III при тестировании сетей на частотах до 300 МГц, что превышает требования для категории 6 (Class E) испытательных стандартов. Прибор поддерживает до 15 различных наборов тестов, что позволит провести глубокий диагностический анализ, включая возможность определения местоположения неисправности. Позволяет переносить результаты измерений или генерируемые профессиональные отчеты на ПК благодаря совместимому ПО. Опционально поставляемые дополнения предоставляют возможность тестирования одномодового и многомодового оптоволокна, а наличие речевого комплекта обеспечит дополнительные удобства в работе. Стоимость – от $7800 до $8500 [4].


Рис. 12. Тестер Acterna LT 8600

Рефлектометр MTS 8000 – новая мультимодульная тестовая платформа для оптоволоконных систем. В этом приборе одновременно инсталлирован рефлектометр, оп-

тический тестер, измеритель оптической мощности, локатор визуальных дефектов, оптический микроскоп, оптическая гарнитура, OTDR. Конструктивное решение, разработанное специалистами Acterna, позволяет одновременно устанавливать в MTS 8000 большое количество сменных оптических модулей, благодаря чему пользователь получает возможность измерения всех необходимых характеристик в зависимости от типа работ. Процессор, установленный в MTS 8000, позволяет тестировать сеть по заранее предустановленным наборам тестов. Внутренняя память устройства составляет 8 МБ. Новой интересной особенностью является возможность установки жесткого диска емкостью до 6 ГБ. Для удобства и возможности оперативной работы в MTS 8000 установлены накопители FDD, CD-RW, а также USB-порты [4].


Рис. 13. Рефлектометр MTS 8000

**Продукты Agilent Technologies.** Переносные кабельные анализаторы WireScope 350 и FrameScope 350 предназначены для администраторов сетей. Усовершенствованные технологии тестирования позволяют свести к минимуму влияния испытательных адаптеров и добиться высокого уровня точности (превышает требования уровня III).


Рис. 14. Рефлектометр MTS 8000

Имея дружественный пользовательский интерфейс, сенсорный LCD-дисплей, диалоговые руководства и встроенный предопределенный набор тестов, прибор позволяет выполнять ряд испытаний, нажимая лишь одну кнопку. Прибор сертифицирует установленные локальные кабельные системы на соответствие проектным спецификациям стандартов категории 6 (Class E). Полезной является возможность обновления аппаратной и программной части для проведения испытаний систем более высоких кате-

горий. С помощью опциональной приставки Fiber SmartProbe можно тестировать оптоволоконные системы (как одномодовые, так и многомодовые). Результаты испытаний прибор сохраняет на сменные карты флэш-памяти. Прибор автоматически составляет профессиональные отчеты, используя программное обеспечение Scope Data Pro, входящее в комплект тестера. Используя дистанционное управление устройством, можно генерировать и сразу же отсылать отчеты о работе сети, например, в центральный офис. Использование гарнитуры позволяет координировать действия в real-time-режиме. Цены для этих устройств колеблются от $7000 до $14000 [4].

## Заключение

Измерения в СКС выполняются на всех этапах строительства и эксплуатации кабельной системы и являются необходимым условием обеспечения нормального функционирования и быстрого восстановления работоспособности каналов и трактов в аварийных ситуациях.

В работе были рассмотрены все основные параметры, по которым определяют характеристики кабельной системы в соответствии с созданными стандартами. Приведен математический аппарат для более четкого представления назначения той или иной функций, измеряемой устройством тестирования. Эти данные крайне важны при чтении непосредственно самих тестов кабельных систем, при помощи которых можно сделать четкий и однозначный вывод и дать правильную характеристику установленной системе. Они позволяют корректировать и приводить в соответствие заданные на начальном этапе параметры, что, в свою очередь, продлевает срок эксплуатации СКС.

В то же время, в связи с появлением новых стандартов и классов кабельных систем, следует ожидать разработки следующего поколения тестеров с расширенным частотным диапазоном, а также более жестких требований к качеству структурированных кабельных систем.

## Литература


1. Семенов А.Б., Стрижаков С.К., Сунчелей И.Р. Структурированные кабельные системы. М.: Компания АйТи, ДМК Пресс, 2004. 640 с.
2. Самарский П.А. Основы структурированных кабельных систем. М.: Компания АйТи, ДМК Пресс, 2005. 216 с.
3. http://www.adp.ru/passive/teh_doc/for_test.htm
4. http://www.cnts-net.ru/stati/statia40.dhtml

# ПЕРСПЕКТИВЫ ПРИМЕНЕНИЯ ОПТИЧЕСКОГО ВОЛОКНА В СИСТЕМАХ УПРАВЛЕНИЯ

**О.В. Елисеев, Д.В. Соловьёв, В.Н. Фролков**
**Научный руководитель – д.т.н., профессор Ю.А. Гатчин**

В статье рассматриваются перспективные характеристики оптического волокна, применимые в таких сферах, как управление критическими системами. Рассматриваются две критические системы управления - лазерная система для осуществления термоядерного синтеза и система управления нестационарными объектами.

## Введение

В основных направлениях экономического и социального развития ставится задача развивать производство электронных устройств регулирования и телемеханики, исполнительных механизмов, приборов и датчиков систем комплексной автоматизации сложных технологических процессов, агрегатов, машин и оборудования. Для осуществления автоматического управления создается система, состоящая из управляющего объекта и тесно связанного с ним управляющего устройства. Как и всякое техническое сооружение, систему управления стремятся создать как бы конструктивно жесткой, динамически «прочной». Изменение автоматически управляемых систем, связанные с повышением интенсивности процессов, усложнения структуры и повышением требований, предъявляемых к скорости протекания, точности и качеству процессов, приводят к необходимости создания более эффективных аналитических методов исследования систем.

Совокупность средств управления и объекта образует системы управления. Система, в которой все рабочие и управляющие операции выполняются автоматическими устройствами без участия человека, называются автоматической системой (АСУ). Круг объектов и операций управления весьма широк. Он охватывает технологические процессы и агрегаты, группы агрегатов, цехи, предприятия, человеческие коллективы, организации и т.д.

В настоящее время все большее распространение получают системы управления на основе оптического волокна. Это можно объяснить тем, что оптические волокна имеют ряд несомненных преимуществ, таких как:

- широкополосность (до нескольких десятков терагерц);
- малые потери (минимальные – 0,154 дБ/км);
- малый (около 125 мкм) диаметр;
- малая (приблизительно 30 г/км) масса;
- эластичность (минимальный радиус изгиба 2 мм);
- механическая прочность (выдерживает нагрузку на разрыв примерно 7 кг);
- отсутствие взаимной интерференции (перекрестных помех типа известных в телефонии «переходных разговоров»);
- безындукционность (практически отсутствует влияние электромагнитной индукции, следовательно, и отрицательные явления, связанные с грозовыми разрядами, близостью к линии электропередачи, импульсами тока в силовой сети);
- взрывобезопасность (абсолютная неспособность волокна быть причиной искры);
- высокая электроизоляционная прочность (например, волокно длиной 20 см выдерживает напряжение до 10000 В);
- высокая коррозионная стойкость, особенно к химическим растворителям, маслам, воде.

Введем понятие критической системы управления. Стремительное расширение сфер внедрения вычислительной техники охватило и так называемые АСУ критическо-

го применения (КСУ), представляющие собой АСУ критическими объектами. К таким объектам можно отнести военные объекты, экологически опасные производства, атомные станции, объекты транспорта, связи, финансово-кредитные сферы и т.д. Критические объекты характеризуются тем, что размеры ущерба или других последствий, которые могут возникнуть в результате нарушения их работоспособности, сбоев и отказов в работе, оказываются неприемлемыми для общества. В связи с этим в КСУ на первый план выходят задачи обеспечения надежности их функционирования.

Рассмотрим теперь несколько моментов практического применения оптического волокна в критических системах управления, а именно в лазерной системе для осуществления термоядерного синтеза и в системах управления нестационарными объектами.

### Лазерная система для осуществления термоядерного синтеза

Заинтересованность в лазерах как возможных источниках энергии возникает в связи с особенностями импульсного высвобождения энергии из мишени с термоядерным топливом без внешнего удержания [1].

Система с помощью компьютеризированной синхронии должна обеспечить передачу 192 лучей NIF длиной 20 нс по 1-километровой оптической дорожке и их прибытие в пределах 30 пс в центр камеры мишени 10 м в диаметре. Основной задачей является сосредоточение полной энергии 1,8 МДж на 2-милиметровой капсуле, содержащей дейтерий и тритий с точностью 50 мкм [2]. Схема системы представлена на рис. 1.

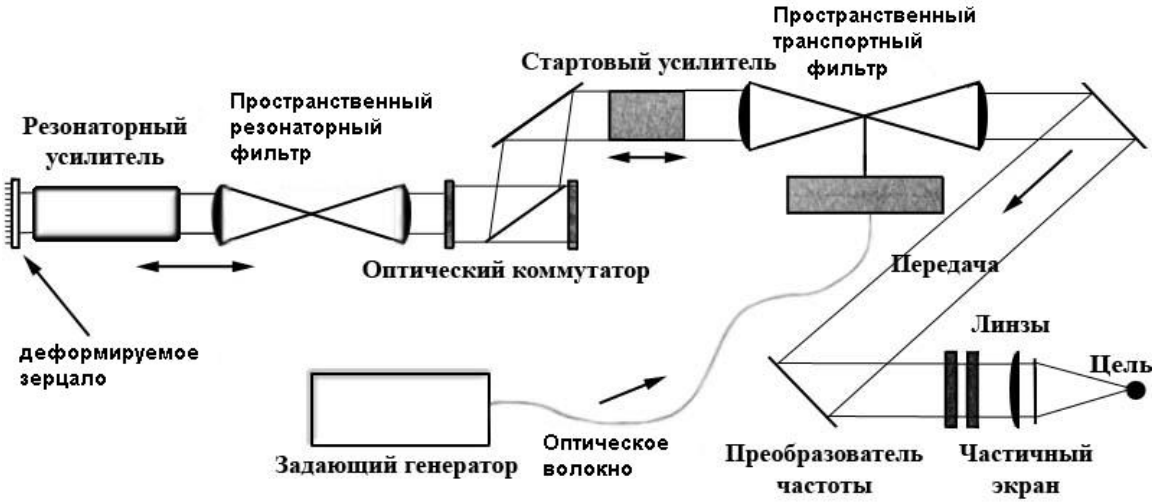



Рис. 1. Схема передачи лазерных импульсов (351 нм)

| Характеристики волокна | Предъявляемые требования |
|---|---|
| Тип профиля | градиентное |
| Временная дисперсия (351 нм) | 0,4 пс/м |
| Затухание (351 нм) | 150 дБ/км |
| Числовая апертура (351 нм) | 0,135±0,01 |
| Диаметр сердцевины | 435 мкм |
| Диаметр оболочки | 590 мкм |
| Диаметр покрытия | 760 мкм |

Таблица 1. Требования к передаточным характеристикам волокна NIF с высокой пропускной способностью, обладающего высокой прозрачностью в УФ

Решение задач диагностики лазерных импульсов в ультрафиолетовой области спектра потребовало создания оптических волокон с минимально возможным поглощением в области 351 нм и минимально искажающим форму импульса (см. табл. 1.). Оптические волокна, обладающие высокой пропускной способностью и низким затуханием, должны передавать ультрафиолетовые лазерные диагностические сигналы длительностью в долю наносекунды из камеры мишени до контролирующих приборов, расположенных в смежных комнатах, с помощью нескольких волокон с длиной порядка 65 м. Для выполнения этой задачи были разработаны и изготовлены специальные оптические волокна, так как никакие существующие доступные волокна не обладают всеми требуемыми характеристиками. Они обладают большим диаметром сердцевины (435 мкм), имеют оптическую дисперсию меньше 0,9 пс/м и затухание менее 150 дБ/км на длине волны 351 нм [3].

На рис. 2. показана форма сигнала на длине волны 351 нм, полученного экспериментально в высокоточной системе контроля, переданного через волокно длиной 25 м.

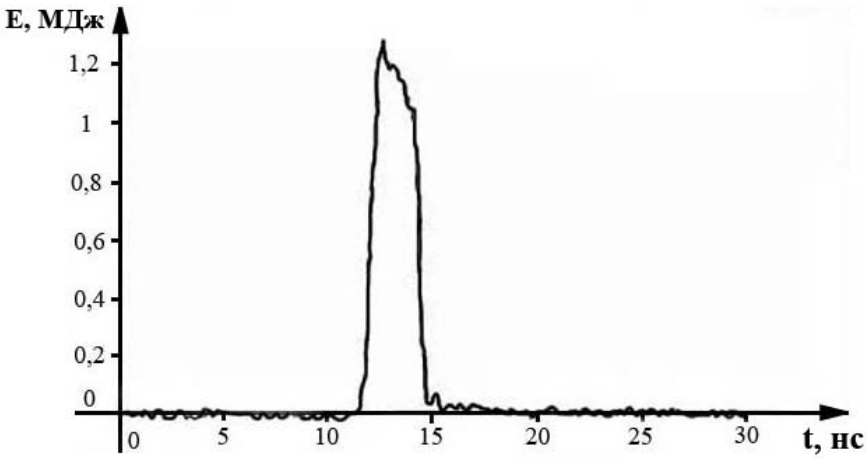



Рис. 2. Типичный сигнал на длине волны 351 нм, полученный датчиком контроля энергии в высокоточной системе контроля после прохождения 25 м волокна

**Применение волокна в системах управления нестационарными объектами**

По мере проведения исследований и разработок методов создания волокон с малыми потерями и высокой прочностью на разрыв становится реальным применение систем с разматываемым кабелем. Команды управления передаются по этому кабелю к летательному аппарату, а с него на наземный пункт передается видеоинформация, на основе которой вырабатываются команды. В результате получается помехозащищенная система связи с летящим объектом, не требующая прямой видимости (рис. 3). К основным требованиям, предъявляемым к этим кабелям, относятся: малый диаметр, малая масса, высокая прочность, широкая полоса пропускания, низкий уровень затухания и возможность эксплуатации в экстремальных условиях [4].

Известно, что стеклянные волокна, находящиеся под растягивающей нагрузкой, ведут себя как линейные упругие твердые тела до тех пор, пока внутреннее напряжение в некоторой точке тела не достигнет критического значения, при котором наступает хрупкий разрыв в этой точке. Поскольку напряжение наиболее сильно в вершинах трещин и дефектах, прочность волокна зависит от их распределения. Конструкция кабеля должна обеспечивать защиту стеклянных волокон от повреждения во время изготовления, прокладки и эксплуатации и удовлетворять требованиям на изгибы, удары, давления (см. табл. 2.). Для защиты волокна от растягивающих напряжений, превышающих допустимые, в кабеле необходимо предусматривать упрочняющие элементы. Материал и количество этих элементов выбирается из расчета достаточной жесткости при растяжении в пределах малого до-

пустимого удлинения световодов [3]. Поперечное сечение волоконно-оптического кабеля для управления летательным аппаратом представлено на рис. 4.

| Характеристики волокна | Особо прочное волокно | Обычное волокно |
|---|---|---|
| Тип профиля | градиентное | градиентное |
| Затухание (1,3 мкм) | 2,3 дБ/км | 0,5 дБ/км |
| Радиационная стойкость | $10^8$ рад | $10^5$ рад |
| Диаметр сердцевины | 35 мкм | 50 мкм |
| Диаметр оболочки | 125 мкм | 125 мкм |
| Диаметр покрытия | 400 мкм | 250 мкм |
| Усилие на разрыв | 60 Н | 7 Н |
| Минимальный радиус изгиба | 3 мм | 25 мм |
| Ширина полосы пропускания | 450 МГц·км | 250 МГц·км |

Таблица 2. Сравнительные характеристики особо прочных волокон, используемых для управления летательными аппаратами

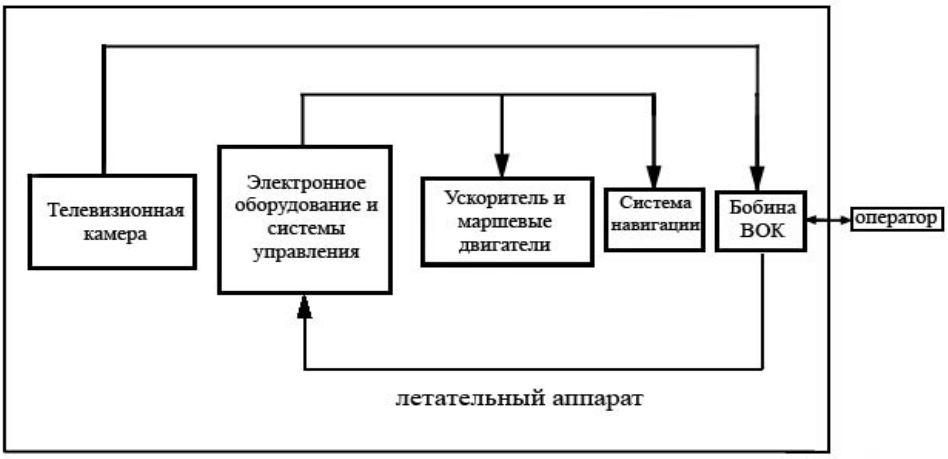



Рис. 3. Система управления летательным аппаратом на основе особо прочного ВОК

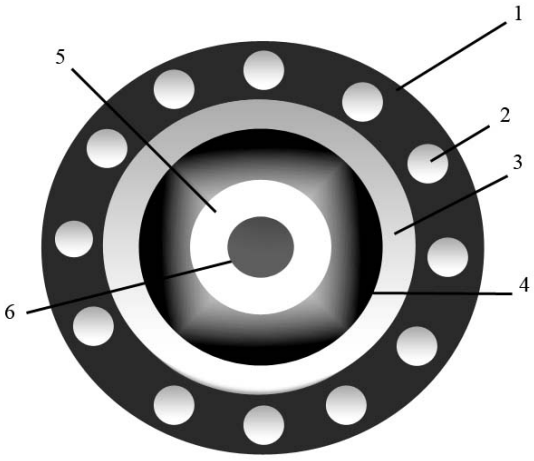



Рис. 4. Поперечное сечение волоконно-оптического кабеля для управления летательным аппаратом: 1 – полиамидное покрытие (400 мкм), 2 – усиливающие терлоновые нити (12 мкм), 3 – эпоксиокрилатное покрытие (230 мкм), 4 – силиконовое покрытие (200 мкм), 5 – оболочка волокна (125 мкм), 6 – сердцевина волокна (35 мкм)

## Заключение

Оптическое волокно находит широкое применение в системах управления, в том числе и в критических системах управления. Это обусловлено неоспоримыми преимуществами по характеристикам оптического волокна по сравнению с кабельными или радиопередающими системами. Изготовление специализированного волокна, подходящего под конкретные условия эксплуатации, является вполне возможным и перспективным направлением при решении задач увеличения надежности критических систем управления.

## Литература


1. Бранер К., Джорна С. Управляемый лазерный синтез. М., Атомиздат, 1977. 144 с.
2. Андреев А.А. Генерация и применение мультитераваттных лазерных импульсов. / Труды ГОИ. Т. 84. В. 218. 2000. С. 21–39.
3. Dukel'sky K.V.., Kondrat'ev Y.N. and others. Low-dispersion optical fiber highly transparent in the UV spectral range. Opt. Engineering, 2004, v. 43, n. 12, p. 2896–2903.
4. Барноски М.К. Волоконно-оптические системы для военных применений. // ТИИЭР. 1980. Т. 68. № 10. С. 180–186.
5. Шварц М.И., Гейген П.Ф., Сантана М.Р. Проектирование и основные характеристики световодного кабеля. // ТИИЭР. 1980. Т. 68. № 10. С. 54–60.

# РАСШИРЕНИЕ ВОЗМОЖНОСТЕЙ ИНТЕРАКТИВНЫХ ПОЛЬЗОВАТЕЛЬСКИХ ИНТЕРФЕЙСОВ WEB-ПРИЛОЖЕНИЙ С ПОМОЩЬЮ ТЕХНОЛОГИИ AJAX

**В.В. Власов**
**Научный руководитель – к.т.н., доцент Б.А. Крылов**

Описаны решения и технологии, которые могут помочь разработчику интерфейсов WEB-приложений создавать мощные высокоскоростные инструменты взаимодействия между клиентом (браузером) и сервером. AJAX – одно из самых перспективных направлений развития интернет-технологий, которое дает принципиально новые возможности WEB-интерфейсу, увеличивает скорость работы пользователя и уменьшает время ожидания получения или обработки информации.

## Введение

О технологии AJAX впервые заговорили после появления 18 февраля 2005 г. статьи Джесси Джеймса Гарретта (Jesse James Garrett) «AJAX – новый подход к разработке веб-приложений» (AJAX: A New Approach to Web Applications) [1]. Автором статьи описывается новая развивающаяся технология, которая является одновременно революционной в подходах к разработке веб-интерфейсов и объединением нескольких самостоятельных технологий, которые могут быть эффективно использованы вместе. AJAX – сокращение от *Asynchronous JavaScript + XML* (Асинхронный JavaScript и XML) – представляет собой фундаментальный сдвиг в сторону увеличения возможностей веб-приложений. Поэтому принято считать, что AJAX – это не самостоятельная технология, а скорее идея.

В последнее время выходит множество различных статей и книг, где рассказывается о совместном использовании AJAX и популярного скриптового языка программирования PHP. В данной статье мы этого вопроса касаться не будем.

Было бы неправильно сказать, что java-разработчики обделены информацией по использованию технологии AJAX, однако большая часть документации доступна на английском языке. Существуют также книги, полностью посвященные AJAX и шаблонам проектирвоания приложений AJAX на стороне клиента [2]. В данной статье будут описаны общие принципы и подходы к проектированию динамических веб-интерфейсов с использованием AJAX, а также практическое применение AJAX с Java 2 Enterprise Edition (J2EE), основанное на документации, предоставляемой Sun Microsystems [3].

## Что такое AJAX

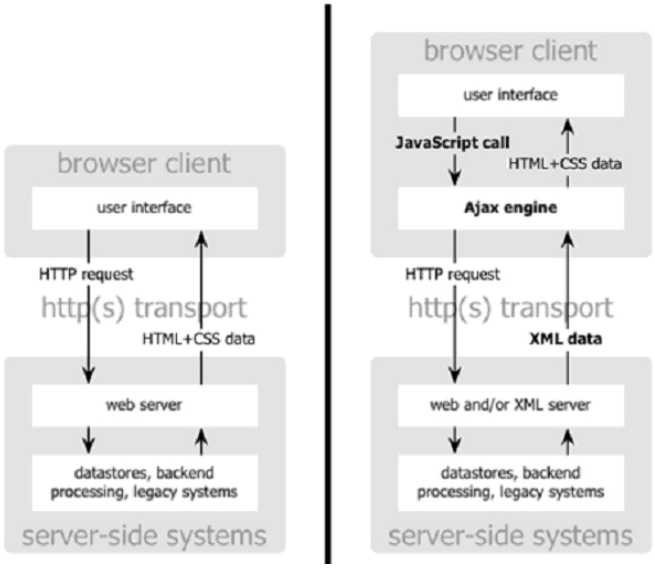
Классические WEB-приложение работает примерно следующим образом: большинство действий пользователя (формы, гиперссылки) вызывает обращение к серверу. В основном используются формы для заполнения и отправки некоторых данных. Пользователь вводит некоторые данные, нажимает Submit. На сервер отправляется эта информация (запрос). Сервер занимается обработкой запроса – принимает данные, выполняет необходимые операции, обращаясь к другим методам, классам и, в конце концов, формирует HTML страничку, которую возвращает пользователю. Подобный подход изначально разрабатывался для того, чтобы использовать WEB как хранилище гипертекстовых документов, однако то, что делает WEB хорошим для хранения гипертекстовых документов, не обязательно делает его хорошим для создания WEB-приложений.

Что делает пользователь, пока сервер выполняет свою работу? Правильно. Он ждет. И, с выполнением каждой следующей операции, он ждет еще и еще. Поэтому разработчики WEB-интерфейсов иногда испытывают зависть к разработчикам оконных

приложений, которые обладают более богатыми возможностями и отзывчивостью при взаимодействии с пользователем, что, как кажется, недостижимо для WEB-приложений. Та же самая простота, которая способствовала быстрому распространению WEB, создает непреодолимое расстояние между тем, что мы можем предложить пользователям WEB-приложений, и тем, что пользователь получает при использовании standalone приложений.

AJAX предназначен для того, чтобы решить эту проблему, расширить возможности интерактивных пользовательских WEB-интерфейсов и приблизить WEB-приложения по функциональности к привычным всем нам GUI-приложениям. Если весь интерфейс уже загрузился, почему взаимодействие пользователя с программой должно прерываться, как только программе понадобились некоторые данные с сервера? Почему пользователь вообще должен видеть, что приложение обращается к серверу? AJAX позволяет сделать взаимодействие приложения с сервером незаметным для пользователя и перезагружать не всю страницу, а только отдельные ее фрагменты.

Идея AJAX довольно проста. Сравним механизмы взаимодействия клиент-сервер (рис. 1). Если в случае обычных WEB-приложений клиент (*browser client*) отправляет по протоколу HTTP запрос на сервер, на сервере происходит обработка запроса, возвращается ответ, страница у клиента в браузере перезагружается и отображает новую страницу, полученную с сервера.




а б

Рис. 1. Сравнение традиционной модели WEB-приложения (а) с моделью AJAX приложений (б)

AJAX приложения позволяют избавиться от прерывистого процесса взаимодействия с WEB-приложением благодаря промежуточному слою между пользователем и сервером – движку AJAX. Казалось бы, добавление нового слоя замедлит взаимодействие между пользователем и программой, но на самом деле возникает обратный эффект. Вместо того, чтобы загружать страницу в начале работы приложения, браузер загружает AJAX движок (*AJAX engine*), который написан на JavaScript. Этот движок отвечает за отображение WEB-страницы и за взаимодействие приложения с сервером. AJAX движок позволяет пользователю осуществлять взаимодействие с сервером асинхронно, т.е. независимо от обращения к серверу. Таким образом, пользователю нет необходимости наблюдать чистую страницу в браузере или иконку песочных часов в ожидании, когда сервер что-нибудь сделает.

Теперь рассмотрим, какие технологии имелись в виду, когда мы говорили о том, что AJAX – не самостоятельная технология, а идея.

AJAX подразумевает совместное использование следующих технологий:

- стандартные средства отображения страниц *DHTML* [4];
- механизмы асинхронной передачи данных с сервера с помощью *XMLHttpRequest* [5], хотя на данный момент этот объект не специфицирован в JavaScript технологии, однако он поддерживается всеми наиболее используемыми браузерами;
- динамические средства отображения информации и взаимодействия с пользователем – *Document Object Model* [6];
- обмен данными и их обработка – *XML* и *XSLT* [7];
- *JavaScript*, который объединяет все это вместе.

Динамическое обращение к серверу «на лету», без перезагрузки всей страницы полностью, может быть реализовано также с использованием следующих инструментов:

- через динамическое создание дочерних фреймов [8];
- через динамическое создание тега <script>[9].

Используя JavaScript, WEB-страница может асинхронно обращаться к серверу, получать некоторое требуемое содержимое, представленное в XML-формате, текста, html-содержимого или в формате *JavaScript Object Notation* (JSON) [10]. Затем JavaScript может обновить или изменить используемую Document Object Model html-страницы в соответствии с новыми полученными данными. Взаимодействие страницы с JavaScript основано на отслеживании таких событий, как загрузка документа, клик мышки, изменение фокуса/позиции курсора (мыши), изменение времени и др. AJAX позволяет четко отделить логику представления от отображаемых данных. Для этого html-страница должна быть «разбита» на составные компоненты: базовые и загружаемые. Для реализации подобного взаимодействия AJAX-у требуются различные серверные технологии (server-side). Обычно серверное WEB-приложение отвечает за создание (генерацию) html-страниц для каждого поступившего от клиента запроса. При следующем запросе (обновлении) страницы опять происходит генерация новой страницы и возвращение ее браузеру клиента. «Толстое», более интеллектуальное WEB-приложение ведет себя по-другому: сначала загружается шаблон (контейнер), т.е. некая базовая часть приложения, которая, в свою очередь, отображает содержимое в зависимости от событий, произошедших на странице (клик мышки, перемещение курсора в другое место), используя XML данные, полученные с серверных компонентов приложения (server-side component), и перезагружает только те фрагменты страницы, которые связаны с данным событием (произведенным пользователем действием). Фактически WEB-приложение ведет себя также как обычное оконное (*desktop*) приложение.

Вот некоторые возможные применения для AJAX:

1. **Валидация данных формы в режиме реального времени (Real-time form data validation).** Под валидацией обычно понимается не просто корректность / правильность типа данных (возраст может быт только числом, но не может быть строкой, если требуется ввести дату, то она должна быть в правильном формате, и т.д.), но и согласованность с определенными критериями. Например, логин пользователя должен быть уникальным (это свойство системы), формат электронного адреса e-mail должен не просто быть строкой, но отвечать определенным критериям, т.е. содержать определенные элементы. Режим real-time подразумевает, соответственно, возможность проверять данные на ходу, не ожидая, пока пользователь произведет какое-либо действие (подтверждение заполнения формы). К данным, которые не могут быть проверены на стороне клиента, можно отнести ID пользователя, серийный номер, пароль, почтовый код и другие данные, которые проверяются на стороне сервера посредством обращения к базе данных или другим хранилищам данных. Очевидно, что в случае обычного WEB-приложения пользователю необходимо отправить форму, прежде чем получить ответ.

2. **Загрузка по запросу (Load on demand).** Этот вид взаимодействия основан на некоем событии, произошедшем на стороне клиента. WEB-приложение позволяет получать дополнительные данные только после запроса (если таковой произойдет), а не загружать все и сразу. Это существенно ускоряет загрузку страницы.
3. **Усовершенствованный интерфейс пользователя и различные эффекты (Sophisticated user interface controls and effects).** Возможность управления такими элементами, как деревья (каталогов), меню, таблицы данных, расширенные текстовые редакторы, календари, загрузки и многое другое без дополнительной перезагрузки страницы делает интерфейс пользователя дружественным и понятным (*user-friendly*).
4. **Обновление и синхронизация данных (Refreshing data and server push).** WEB-приложения могут получать актуальные (своевременные) данные, такие, как состояние счета, котировки акций, погода, голосования и другие специфические для вашего приложения данные. Однако не всегда эффективно использовать для этих целей AJAX. Такие технологии, как Comet [11], разработаны для того, чтобы сохранять постоянное соединение между клиентом и сервером для получении актуальных данных, которые могут меняться до нескольких раз в секунду.
5. **Частичное подтверждение (Partial submit).** WEB-приложение может принять на подтверждение не полностью (частично) заполненные данные формы без полного обновления страницы.
6. **Объединение данных (Mashes).** WEB-приложение может получать данные, используя прокси со стороны сервера или внешние скрипты, чтобы смешивать данные с внешних (других) источников с данными вашего приложения и служебными данными. Это дает возможность создавать свои собственные сервисы. Например, вы можете смешивать данные сторонних источников (таких как Google Maps) с вашими собственными.
7. **Страница как приложение (Page as an application).** AJAX технология позволяет создавать вам свои собственные одностраничные приложения, которые имеют интерфейс desktop приложений. Например, совмещение технологий AJAX и портлетов (portlets) [12], проект Bindows [13].

Этот список можно продолжать дальше, однако и этих примеров достаточно, чтобы показать, что применение AJAX расширяет возможности интерактивных пользовательских WEB-приложений и ограничено только воображением разработчика (архитектора приложений).

## Анатомия AJAX приложения на примере взаимодействия с J2EE платформой

Рассмотрим на конкретном примере, как работает WEB-приложение с AJAX на клиентской стороне и Java на стороне сервера. Наше WEB-приложение будет очень простое, правда все равно немного сложнее, чем Hello World! Для его реализации вам понадобится сервер приложений (например, JBoss [14]), JRE 1.5, SDK 1.5, библиотека ANT [15] для сборки (War-архива) и развертывания приложения, а также знания упоминаемых технологий.

Наше приложение содержит статическую HTML-страницу (либо сгенерированную сервером JSP страницу – подробнее о JSP технологии см. [16]). Страница содержит форму, которая заполняется некоторыми данными. Эти данные проверяются без перезагрузки страницы. Серверный компонент – сервлет (*servlet*) [17] называется `ValidateServlet`, он обеспечивает проверку формы на стороне сервера (бизнес логика приложения). Рис. 2 детально описывает процесс взаимодействия клиента и сервера. Весь процесс работы приложения можно разбить на несколько этапов (рис. 2).

1. На стороне клиента происходит некоторое событие, вызывается обработчик событий.

2. Создается и настраивается *XMLHttpRequest* объект.
3. XMLHttpRequest объект посылает запрос к серверу.
4. Запрос (*request*) обрабатывается сервером (в нашем случае сервлетом *ValidateServlet*).
5. ValidateServlet возвращает *XML* документ (*response*).
6. XMLHttpRequest вызывает callback() функцию и обрабатывает результат.
7. HTML DOM обновляется.

Рассмотрим каждый шаг взаимодействия с AJAX подробно.

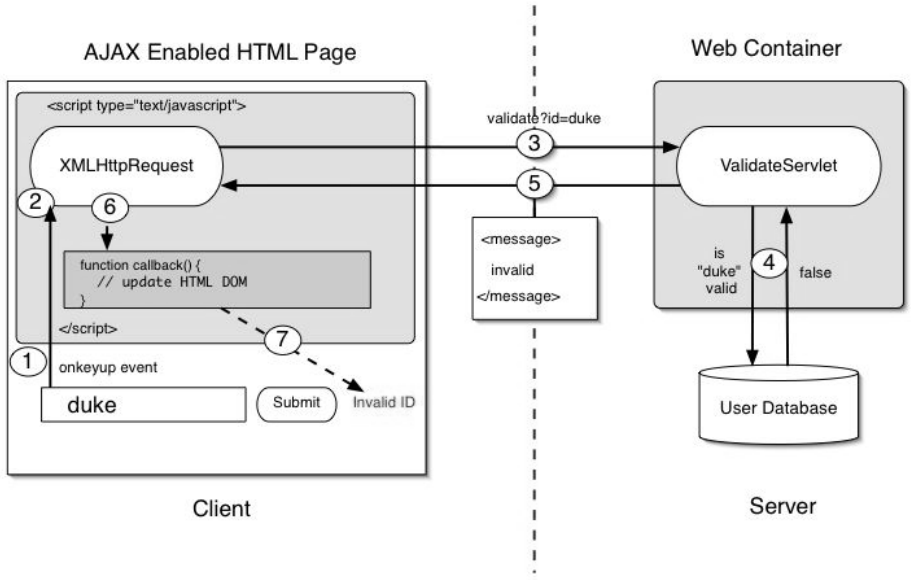



Рис. 2. Взаимодействие клиента и сервера с использованием AJAX на стороне клиента и Веб-контейнера J2EE платформы

1. **Возникновение события.** В ответ на некоторое событие (заполнение формы) вызывается функция JavaScript. В нашем случае функция `validate()` обрабатывает событие `onkeyup`, относящееся к ссылке или компонента формы.

```
<input type="text" size="20 id="userid" name="id"
 onkeyup="validate();">
```

Элемент формы вызывает функцию `validate()` каждый раз, когда пользователь нажимает кнопку в форме.

2. **Конфигурирование XMLHttpRequest объекта.** Создается новый экземпляр объекта XMLHttpRequest.

```
var req;

function validate() {
 var idField = document.getElementById("userid");
 var url = "validate?id=" + encodeURIComponent(idField.value);
 if (typeof XMLHttpRequest != "undefined") {
 req = new XMLHttpRequest();
 } else if (window.ActiveXObject) {
 req = new ActiveXObject("Microsoft.XMLHTTP");
 }
 req.open("GET", url, true);
 req.onreadystatechange = callback;
 req.send(null);
}
```

Функция `validate()` создает `XMLHttpRequest` объект. Метод этого объекта `open()` требует три параметра: HTTP метод (`GET` или `POST`); URL серверного компо-

нента, обрабатывающего данное событие – в нашем случае URL содержит имя сервлета с параметрами и переменную типу boolean, обозначающей должно ли взаимодействие быть асинхронным. API будет следующим `XMLHttpRequest.open(String method, String URL, boolean asynchronous)`. Если взаимодействие должно быть асинхронным (`true`), то должна быть определена функция `callback()`. Функция `callback()` определяется выражением `req.onreadystatechange = callback;`. Более подробно это будет описано в п. 6.

3. **XMLHttpRequest объект посылает запрос к серверу.** Вызывается `req.send(null);` Если запрос отправляется методом `GET`, содержимое может быть пустым или `null`. Браузер отправляет запрос на сервер, используя URL, который был сконфигурирован для объекта `XMLHttpRequest`. В нашем случае данные присоединены к запросу в виде значение=параметр.

Используйте `GET` для «неполноценного» запроса, когда два одинаковых запроса возвращают одинаковые результаты. Учтите, когда используется запрос `GET`, длина строки URL (включая специальные символы) ограничивается некоторыми браузерами и контейнерами сервлетов. Метод `POST` используется, когда посылаемые данные необходимо скрыть из строки адреса URL. Метод `POST` требует `Content-Type` заголовка, который необходимо установить следующим образом:

```
req.setRequestHeader("Content-Type", "application/x-www-form-
urlencoded");
req.send("id=" + encodeURIComponent(idTextField.value));
```

Функция `encodeURIComponent()` обеспечивает корректную передачу данных в зависимости от локализации и отображает специальные символы, в том числе кириллические, в том виде, в каком они пригодны для HTTP запроса.

4. **Обработка запроса на стороне сервера.** Сервлет `ValidateServlet` вызывается контейнером, когда на сервер приходит обращение, содержащее URI «validate» («маппинг» (*mapping*) настраивается в конфигурационном xml файле вашего веб-приложения, например, web.xml).

Ниже показан код сервлета, извлекающего `id` параметр из запроса request и выполняющий всю бизнес-логику на стороне сервера.

```
public class ValidateServlet extends HttpServlet {

 private ServletContext context;
 private HashMap users = new HashMap();

 public void init(ServletConfig config) throws
 ServletException {
 super.init(config);
 this.context = config.getServletContext();
 users.put("greg","account data");
 users.put("duke","account data");
 }

 public void doGet(HttpServletRequest request,
 HttpServletResponse response)
 throws IOException, ServletException {

 String targetId = request.getParameter("id");

 if ((targetId != null) &&
 !users.containsKey(targetId.trim())) {
 response.setContentType("text/xml");
 response.setHeader("Cache-Control", "no-
```

```
 cache");
 response.getWriter().write("<message> valid
 </message>");
 } else {
 response.setContentType("text/xml");
 response.setHeader("Cache-Control", "no-
 cache");
 response.getWriter().write("<message> invalid
 </message>");
 }
 }
}
```

В нашем примере не преследуется цель создания кода, соответствующего «правилам хорошего тона» в программировании, а просто показаны базовые операции. Метод `init()` инициализирует сервлет. В этом есть необходимость только в данном примере. В этом методе создается `HashMap`, содержащий пользователей. Теоретически у нас должно быть некое хранилище данных (база данных), в котором хранится ID клиента, требующее проверки. На практике чаще всего используются технологии, позволяющие абстрагироваться от необходимости писать SQL-команды (или запросы к БД) в коде, основанные на бинах (*Beans*), например, EJB [18]. Эта технология позволяет разработчику не заботиться о типе БД (MySQL, Oracle, MSSQL и др.), самостоятельно управляет транзакциями, обеспечивает масштабируемость приложения и много полезных вещей. Также зарекомендовала себя ORM технология Hibernate [19], которая выполняет те же функции, однако более проста и конфигурируема при разработке.

В методе `doGet()` выполняется проверка, что параметр `targetId` существует и его значение (имя пользователя) не содержится в коллекции `users`.

5. **`ValidateServlet` возвращает ответ клиенту.** В зависимости от результата проверки `if/else`, отправляется ответ в формате XML. Пользователь с ID `duke` присутствует в нашей коллекции, поэтому возвращается XML документ, содержащий только один элемент (тэг) message со значением `invalid`.

Разработчик должен учесть следующие детали. Во-первых, `Content-Type` должен быть установлен `text/xml`. Во-вторых, `Cache-Control` должен быть `no-cache`. `XMLHttpRequest` объект обрабатывает только те запросы, которые удовлетворяют этим условиям. `no-cache` означает, что браузер клиента кэширует ответы сервера в том случае, если с одинаковых адресов на один и тот же запрос приходят разные ответы.

6. **`XMLHttpRequest` вызывает `callback()` и обрабатывает результат.** `XMLHttpRequest` сконфигурирован так, что после получения ответа с сервера вызывается функция `callback()`. Проверяется, что запрос выполнен `readyState==4` и код ответа 200 (успешно).

```
function callback() {
 if (req.readyState == 4) {
 if (req.status == 200) {
 // update the HTML DOM based on whether or not message is valid
 }
 }
}
```

Браузеры поддерживают объектное представление информации. HTML-страница имеет доступ к DOM модели, и API позволяет JavaScript модифицировать DOM после обновления данных.

Когда клиенту возвращается успешный запрос, можно использовать XMLHttpRequest.responseXML. DOM API дает возможность JavaScript управлять содержимым страницы. Следующая функция делает «разбор» XML-документа. Вызов функции setMessage() обновляет DOM.

```
function parseMessage() {
 var message =
req.responseXML.getElementsByTagName("message")[0];
 setMessage(message.childNodes[0].nodeValue);
}
```

7. **HTML DOM обновлен.** JavaScript может обратиться к любому элементу на странице, используя идентификаторы. Функция document.getElementById("userIdMessage") возвращает ссылку на элемент, который отмечен идентификатором "userIdMessage". Используя эту ссылку, можно изменить атрибуты элемента, свойства стиля, добавить, удалить, изменить дочерние элементы. Содержимое тела элемента может быть изменено вызовом свойства innerHTML, как в следующем примере:

```
<script type="text/javascript">
...
function setMessage(message) {
 var mdiv = document.getElementById("userIdMessage");
 if (message == "invalid") {
 mdiv.innerHTML = "<div style=\"color:red\">Invalid
 User Id</ div>";
 } else {
 mdiv.innerHTML = "<div style=\"color:green\">Valid
 User Id</ div>";
 }
}
</script>
<body>
<div id="userIdMessage"></div>
</body>
```

Часть HTML-страницы будет изменена. Если свойство innerHTML содержит такие элементы, как <image> или <iframe>, то содержимое этих элементов загрузит новые данные с сервера и будет обновлено. Такие AJAX приложения, как Google Maps [20], реализуют технологию добавления изображений, используя вызовы AJAX, чтобы динамически строить карту.

## Заключение

Самый большой прорыв в использовании AJAX подхода не является техническим. Основные технологии, используемые в AJAX, уже давно используются, стабильны и хорошо осмыслены. Прорыв для разработчиков WEB-приложений заключается в том, чтобы перешагнуть через общепринятые ограничения и быть открытыми для более широких и богатых возможностей. Уже сейчас пользователь Интернета может использовать приложения, построенные на идеологии AJAX: Flickr, GMail, Google Suggest или Google Maps. А что будет завтра? Системы автоматизированного проектирования в Интернет? On-line операционные системы, загружаемые с сервера производителя? Возможности, которые дает идеология AJAX, по истине безграничны.

## Литература


1. J. J. Garrett. AJAX: A New Approach to Web Applications (http://www.adaptivepath.com/publications/ essays/archives/000385.php), 2005.
2. Крейн Д., Паскарелло Э. Ajax в действии. М.: Изд. дом «Вильямс», 2006.
3. G. Murray, Asynchronous JavaScript Technology and XML (AJAX) With the Java Platform. (http://java.sun.com/developer/technicalArticles/J2EE/AJAX/), 2006.
4. W3C: Dynamic Accessible Web Content Roadmap (http://www.w3.org/WAI/PF/roadmap/ DHTMLRoadmap040506.html), April 5, 2006.
5. W3C: The XMLHttpRequest Object (http://www.w3.org/TR/XMLHttpRequest/), 27 February 2007.
6. W3C: Document Object Model (www.w3.org/DOM/ ), 2005.
7. W3C: XSL Transformations (XSLT) Version 1.0 (http://www.w3.org/TR/xslt), 16 November 1999.
8. Remote Scripting with IFRAME (http://developer.apple.com/internet/webcontent/iframe.html ).
9. JsHttpRequest: динамическая подкачка данных с поддержкой AJAX (http://dklab.ru/lib/JsHttpRequest/).
10. JavaScript Object Notation data-interchange format (www.json.org/).

# 5 МЕТОДЫ ЗАЩИТЫ ИНФОРМАЦИИ

## СИСТЕМА АВТОМАТИЗАЦИИ И ОПТИМИЗАЦИИ ПРОПУСКНОЙ СПОСОБНОСТИ ОФОРМЛЕНИЯ ДОКУМЕНТОВ

**Г.С. Александров, О.В. Елисеев, Д.В. Соловьёв, А.А. Федоров**
**Научный руководитель – д.т.н., профессор В.Л. Ткалич**

Представлен обзор предполагаемой системы управления формами документов в различных областях применения.

### Введение

Данный проект предназначен для оптимизации и автоматизации пропускной способности оформления документов. В государственных учреждениях, как и во многих коммерческих, огромное количество времени теряется на оформление документов, таких как выдача справок, заполнение форм и многих других документов. Особенно данная тенденция просматривается в государственных организациях. В настоящем проекте мы планируем добиться уменьшения времени, затраченного на операции оформления документов. Актуальность данной разработки заключается в том, что разрабатываемый комплекс аппаратного и программного обеспечения можно адаптировать под большое количество имеющихся форм заполнения документации.

### Описание проекта

| | |
|---|---|
| *Общее* | |
| Тип матрицы | CIS |
| Тип | офисный |
| *Размеры* | |
| Макс. размер документа в проход. свете | 148x105 мм |
| Максимальный формат оригинала | A6 |
| *Дополнительно* | |
| Вес | 1 кг |
| Особенности | Паспортный сканер для сканирования нестандартных документов. Возможность работать с поднятой крышкой. Автоматическое выравнивание и обрезка изображений. |

Рис. 1. Fujitsu fi-60f, A6, Simplex, Flatbed, USB 2.0

Данный проект состоит из двух частей.

Первая часть реализована в виде сканирующего элемента, который отвечает за получение информации с паспорта. Процесс сканирования занимает примерно одну

или две секунды, что является существенным преимуществом. Сканирование происходит в рамках, заданных программным обеспечением. Происходит сканирование всей страницы, а затем области, нужные для распознавания, определяет программное обеспечение. Технические характеристики представлены на рис. 1.

Вторая часть – это программное обеспечение, отвечающее за все дальнейшие операции системы. К основным операциям можно отнести распознание отсканированного текста, запись и хранение в буфере памяти, высвобождение информации в соответствии с задачей, запись сформированного документа. Алгоритм работы представлен на рис. 2. Разработка программного обеспечения, предположительно, будет реализована на платформе программирования языка C++ Builder 6.0, который облегчает визуальное программирование.

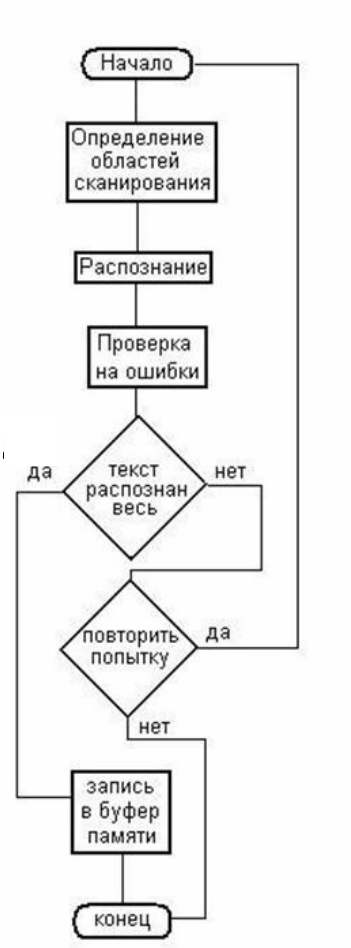



Рис. 2. Алгоритм работы программного обеспечения

Интерфейс пользователя должен быть как можно проще, поскольку работающие в данном комплексе пользователи могут быть не совсем компетентны в области IT-технологий. С помощью интерфейса пользователи (как будет рассказано ниже) смогут сами редактировать конечные шаблоны.

Работа программы начинается с алгоритма распознавания текста переданного со сканера. Обеспечиваются такие области сканирования, как фамилия, имя, отчество, дата и место рождения, номер и серия паспорта, дата и место выдачи паспорта, прописка. Перечисленные области записываются в ячейки памяти, для дальнейшего добавления их в форму запроса.

Форма запроса создается пользователем вручную, по соответствию требований. Форма представляет собой обычный документ Microsoft Word, в котором пользователь

может заполнять поля по своему усмотрению. Возможен процесс импортирования заранее подготовленной формы, сделанной в том же самом формате.

Затем начинается процесс формирования документа с подготовкой для дальнейшей автоматической работы. Пользователь после окончания формирования шаблона начинает расставлять ячейки для добавления распознанных фрагментов, с помощью несложных манипуляций добавляя в созданную им форму ячейки, где в дальнейшем будет располагаться распознанный текст. При дальнейшем сканировании форма автоматически заполняется те поля, куда пользователь изначально расставил поля панели.

Панель пользователя представляет собой простейший набор кнопок, в которых находится информация из сканированной области.

Такой процесс представления удобен для обычных пользователей. Преимущество данной ступени работы программы – в том, что не нужно обладать какими-либо дополнительными навыками, чтобы заполнить форму документа. При дальнейшей работе пользователь просто выбирает сохраненный документ и начинает работу. Последующее заполнение не требует от пользователя никаких дополнительных операций.

Далее при последующих сканированиях пользователь только выбирает форму, которую он сохранил, и программа при нажатии кнопки «Start» автоматически заполняет места, которые указаны в форме.

В тех местах, где пользователь разместил поля заполнения, при дальнейшем сканировании появляется распознанный текст, полученный со сканера.

Процесс происходит автоматически. Сканер получает изображение с паспорта, программа распознает текст и расставляет в те места на странице, которые выбрал пользователь. Если требуется внести какие-либо коррективы, в окне пользователь видит всю страницу и может вносить изменения. После согласия с правильностью заполнения формы документ выводится на печать.

## Заключение

Преимуществом программы является то, что процесс заполнения форм становится печатным и практически автоматизированным. Программа может применяться в любой сфере деятельности. Для работы в программе не требуется дополнительных навыков. Из этого следует, что скорость оформления документов вырастает примерно на 300–400%, что значительно улучшит отношение населения к государственным структурам.

## Литература


1. Стэнли Б. Липпман C++ In-Depth Box Set First Edition, Vol. 1: Essential C++.
2. Послед Б.С. Access 2000. Базы данных и приложения. Лекции и упражнения, 2000.
3. Нечаев В.М. Microsoft Excel. Электронные таблицы и базы данных в задачах. 2001.
4. Рэнди Джей Яргер, Джордж Риз, Тим Кинг. MySQL и mSQL. Базы данных для небольших предприятий и Интернета. 2000.
5. Сергеев А.П. Программирование в Microsoft Visual C++. Самоучитель. 2005.
6. Герб Саттер. Решение сложных задач на C++. Серия книг "C++ In-Depth".
7. Роберт С.М. Быстрая разработка программ на Java и C++: принципы, примеры, практика.
8. Таненбаум Э. Operating Systems: Design and Implementation.
9. Послед Б.С. Borland C++ Builder 6. Разработка приложений баз данных.
10. Калверт Ч. Языки и среды программирования » Borland C++ Builder.

# ОРГАНИЗАЦИЯ БОРЬБЫ С ПРЕСТУПЛЕНИЯМИ В СФЕРЕ БАНКОВСКОГО КРЕДИТОВАНИЯ

**М.В. Лекомцева, В.А. Семенов, М.А. Семенова**
**Научный руководитель – д.т.н., профессор А.Г. Коробейников**

В статье рассматриваются способы совершения преступлений в сфере банковского кредитования, а также показана организация борьбы с ними.

Преступления в банковской сфере условно можно разделить на 2 группы – насильственные преступления (убийства, налеты, грабежи и т.д.) и так как называемые «беловоротничковые» преступления, к которым относятся всевозможные финансовые хищения и махинации т.е. мошенничество. В мировой, как и в российской банковской сфере, группа последних преступлений является явно преобладающей и по количественным показателям, и по объектам похищенных средств.

Мошенничество – это преступление, заключающиеся в завладении чужим имуществом или правом на него, а также в получении иных благ путем обмана или злоупотребления доверием. Наиболее характерным видом мошенничества в банковской сфере является незаконное получение кредита, т.е. получение кредита либо льготных условий кредитования путем предоставления кредитной организации или иному кредитору заведомо ложных сведений о хозяйственном положении либо финансовом состоянии, в связи с чем кредитной организации причиняется крупный ущерб [1].

Проблему защиты банковских интересов от преступных посягательств со стороны недобросовестных заемщиков без преувеличения можно назвать одной из наиболее острых. По предварительным прогнозам и с учетом международного опыта борьбы с преступностью в банковской сфере, этот вопрос и в ближайшие годы останется весьма актуальным. Данное преступление совершается путем обмана. Кредит либо льготные условия кредитования предоставляются потому, что заемщик – руководитель организации или индивидуальный предприниматель – сообщает кредитной организации заведомо ложные сведения о своем хозяйственном состоянии или финансовом положении. На основе анализа и оценки этих объективно неверных данных кредитным комитетом банковского учреждения принимается решение о выдаче денежных средств. Как показывает обобщение банковской и правоохранительной практики, из общего числа невозвращенных кредитов примерно каждый пятый был получен незаконным путем при помощи недобросовестного заемщика и представителя кредитной организации.

Льготные условия кредитования могут выражаться в доверительной выдаче кредита (без обеспечения) или его неполном обеспечении, а также касаться суммы кредита и сроков его возврата; в уменьшении процентной ставки за его пользование. В то же время льготные условия кредитования могут выражаться и в увеличении суммы лимита кредитного риска на конкретного заемщика. Это может произойти при получении ложных сведений, на основании которых делается более благоприятный, чем это объективно следует, вывод об определении группы кредитного риска, т.е. о более высокой вероятности исполнения кредитных обязательств с вытекающими из этого позитивными (для заемщика) последствиями выдачи кредита [2].

Незаконное получение кредита зачастую связано с предоставлением заведомо ложных сведений, которые могут быть как в официальных документах, содержащих необходимые реквизиты (штамп, печать, дата, номер), так и в документах, хотя и не имеющих статуса официальных, но, безусловно, исходящих от заемщика и обладающих основаниями для принятия решения по выдаче кредита.

Заведомо ложные сведения – это неверные данные, о которых заемщик достоверно знает, что искажают или скрывают истинное положение вещей. Заведомая ложность этих сведение состоит в том, что в них осознанно не внесены верные или отражены не-

полные данные, искажающие смысл и содержание представленной информации, в результате чего кредитной организацией делаются неверные оценки в отношении заемщика [3]. Ложные сведения могут вноситься в документы следующими основными способами: собственно внесением в подлинный документ записей, не соответствующих действительности; подделкой документа, которая состоит в изготовлении (составлении) полностью подлинного документа; фальсификацией документа (частичной подделкой), т.е. внесением искаженных сведений в подлинный документ.

В основном к заведомо ложным сведениям относятся информация о хозяйственном положении и финансовом состоянии заемщика. Под хозяйственным положением заемщика обычно понимаются совокупность данных, характеризующих его правовой и экономический статус, его связи, ведения экономической деятельности, обеспеченности деятельности и т.д. К заведомо ложным сведениям о хозяйственном положении можно отнести:

- неверные данные об учредителях, руководителях, акционерах, основных партнерах предприятия, связях с другими фирмами;
- фиктивные гарантийные письма, поручительства, материальные ценности, представление в залог имущества, на которое нельзя обратить взыскание, которое не соответствует объявленной стоимости, не является собственностью залогодателя и т.п.;
- технико-экономическое обоснование (бизнес-план), в котором неверно указаны основные направления использования заемных средств, конкретные хозяйственные операции;
- сфальсифицированные договоры, платежные, транспортные и иные документы о хозяйственной операции, на которую испрашивается кредит;
- поддельные договоры и другие документы, неправильно свидетельствующие о возможности реализации заемщиком своей продукции, его конкурентоспособности, положении на рынке, в отрасли и т.д.;
- данные складского и бухгалтерского учета и другие.

Документы, отражающие финансовое состояние, дают представление не только о денежных средствах заемщика, но о стоимости всех его активов, включающей, кроме денежных средств в кассе, на расчетном счете и у подотчетных лиц, еще такие позиции, как стоимость сырья, готовой нереализованной продукции, стоимость оборудования, зданий сооружений. Из финансовых документов можно почерпнуть сведения о наличии собственных средств и долговых обязательств заемщика [4]. К заведомо ложным сведениям о финансовом состоянии относятся сфальсифицированные: бухгалтерские документы о регистрации в налоговой инспекции, в которых финансовое состояние показано в лучшем положении (баланс-форма №1, отчет-форма №2 и другие), справки о дебиторской и кредиторской задолженности, о полученных кредитах и займах в других банковских учреждениях, выписки из расчетных и текущих счетов и другие.

К основным способам совершения преступлений в сфере банковского кредитования также относятся лжепредпринимательство, преднамеренное и фиктивное банкротство.

Лжепредпринимательство – создание коммерческой организации без намерения осуществлять предпринимательскую или банковскую деятельность, имеющие целью получение кредитов, освобождение от налогов, извлечение иной имущественной выгоды или прикрытие запрещенной деятельности, причинившее крупный ущерб гражданам, организациям или государству. Создание фиктивной коммерческой организации может осуществляться следующими способами:

- использование искаженной или фиктивной информации о предприятии;
- регистрация организации на подставных физических лиц или назначение их на руководящие должности;
- регистрация организации по фиктивному адресу, по недействительным документам или с нарушением закона с помощью подкупа должностных лиц;

- изготовление подложных уставов, регистрационных и иных документов с использованием поддельных печатей, ксерокопий действительных документов;
- использование реквизитов распавшихся предприятий; похищение регистрационных документов чужих предприятий и открытие по ним расчетных счетов и другие приемы [5].

Распространенными приемами при создании фиктивного предприятия являются также: регистрация организации по фиктивному адресу, по недействительным или похищенным документам или с нарушением закона с помощью подкупа должностных лиц, использование реквизитов распавшихся предприятий и т.д.

Преднамеренное банкротство – умышленное создание или увеличение неплатежеспособности с целью невозврата кредита. Фиктивное банкротство – заведомо ложное объявление о своей несостоятельности в целях введения в заблуждение кредитной организации для получения отсрочки или рассрочки причитающихся кредитору платежей или скидки с кредита, а также для его неуплаты [1]. Неправомерные действия при банкротстве заключаются в следующем:

- сокрытие имущества или имущественных обязательств, сведений об имуществе, о его размере, местонахождении либо иной информации о нем, передача имущества в иное владение, отчуждение лил уничтожение имущества;
- сокрытие, уничтожение, фальсификация бухгалтерских и иных учетных документов, отражающих экономическую деятельность;
- неправомерное удовлетворение имущественных требований отдельных кредиторов;
- принятие такого удовлетворения кредитором, знающим об отданном ему предпочтении несостоятельным должником в ущерб другим кредиторам.

Опыт работы кредитных организаций показывает, что немалое количество противоправных посягательств на их кредитные ресурсы могли бы предотвратить сами банковские работники при соответствующей проверке потенциальных ссудозаемщиков и правильной организации этой работы. Однако в большинстве случаев роль подразделений безопасности в этом процессе сведена к минимуму, и их потенциал используется не в полной мере. В некоторых банковских структурах сотрудники подразделений безопасности в полной мере проверяют лишь недобросовестных клиентов или рискованные сделки. Зачастую они привлекаются лишь в тех случаях, если возникла необходимость установить личность клиента, его платежеспособность, надежность. Другое дело, когда возникает ситуация с невозвратом в срок полученного кредита, а у клиента нет имущества и денежных средств для его погашения. К сожалению, имеют место факты, когда руководство кредитной организации поручает подразделению безопасности вернуть этот кредит, не заботясь о том, каким путем это будет сделано. Здесь необходимо четко понимать, что имеющиеся у подразделений возможности и предоставленные им Федеральным законом «О частной детективной и охранной деятельности в Российской Федерации» права ограничены, в связи с чем на завершающей стадии кредитной деятельности, т.е. при невозврате выданного кредита, принять действенные меры к его погашению сотрудники этих подразделений в большинстве случаев не могут.

Работу по проверке клиентов, исходя из основных принципов организации и функционирования комплексной системы обеспечения безопасности кредитной организации, целесообразно организовывать силами трех подразделений (кредитное, юридическое, безопасности), причем каждое из них должно отвечать за решение строго определенных вопросов, отнесенных к их компетенции. В обязанности кредитного подразделения должны входить вопросы проверки и анализа финансово-хозяйственной деятельности предприятия: проверка балансов и других бухгалтерских документов, представленных клиентом, экономическая оценка хозяйственной деятельности потенциального ссудозаемщика, ее эффективность, перспективы и т.п., т.е. его финансовое положение, качество управления компанией, состояние отрасли и региона, конкуренто-

способность клиента, его положение в отрасли или сфере деятельности, платежеспособность, наличие активов и пассивов, товарных запасов, оценка залога и возможность его реализации, а также финансовое положение гаранта или поручителя.

Юридическое подразделение осуществляет следующие функции: проверяет соответствие регистрационных и учредительных документов действующему законодательству, дает заключение о полноте представленных клиентом документов в случае принятия в залог имущества, проводит консультации по вопросу хранения залога, анализирует правильность оформления поручительств и гарантий, проверяет правильность юридического оформления других документов и договоров, в случае невозвращения кредита в срок оформляет документы для предъявления должнику гражданского иска или совершает необходимые действия, связанные с получением исполнительной подписи нотариуса.

К компетенции подразделения безопасности должны быть отнесены вопросы по установлению репутации клиента с позиции возможного противоправного поведения [6]:
1. Технико-криминалистический анализ учредительных и иных документов с целью выявления подделок. Эта работа заключается в проверке:

- соответствия документов общеустановленным формам (наличие необходимых реквизитов; четкость оттисков печатей, штампов; отсутствие разночтений в экземплярах одного и того же документа и подчисток, исправлений, дописок, травлений; соответствие подписи должностных лиц – отсутствие извилистости, угловатости, сдвоенности штрихов, вдавленных бесцветных штрихов);
- в регистрационных, налоговых и иных органах – факта регистрации и постановки на учет и соответствия этих сведений представленным данным;
- в органах милиции – факта утраты паспорта и регистрации по нему предприятия;
- соблюдения требований по созданию организации, в том числе, с участием иностранного капитала. Проверяется, в какие холдинги (финансово-промышленные группы, иные объединения коммерческих организаций) входит организация, имеются ли дочерние и зависимые общества, а также взаимоотношения между организациями – участие в уставных капиталах друг друга, совместное руководство (нахождение одного и того же лица на руководящих постах в разных организациях), совместная хозяйственная деятельность;
- фактического адреса или причины несовпадения юридического и фактического адреса, получение сведений о том, где ранее находилась организация, где она собирается размещаться в дальнейшем, в чьей собственности находится помещение, на какой срок и когда заключен договор аренды, своевременно ли внесена арендная плата, взаимоотношения учредителей с собственником или арендодателем;
- репутации клиента – судимость, психические недостатки, дееспособность, компетентность, отношение к выполнению своих обязательств в прошлом, наличие имущественных претензий и долгов;
- соответствия бухгалтерских данных сведениям из налоговых инспекций;
- достоверности представленных сведений об обеспечении обязательств, причины расхождений между данными складского и бухгалтерского учета и данными об остатках товарно-материальных ценностей, нет ли ареста или иных запретов на предмет залога, в том числе прав третьих лиц, проверка кредитоспособности поручителя, выдачи ли им других поручительств (кому и за что), подлинность банковской гарантии;
- наличия расчетных счетов, которыми может пользоваться заемщик, в том числе расчетных счетов родственных предприятий, особых отношений с предприятиями и лицами, которые могут быть сообщниками клиента и где он может скрыть свое имущество;

- отношений с основными партнерами по приобретению и сбыту сырья, продукции; были ли ранее факты банкротства; каковы учредители и руководители организации; наиболее «тесные и близкие» отношения с партнерами и контрагентами.

2. Проверка факта действительности существования клиента, его репутации.
3. Проверка наличия имущества, предоставленного в залог.
4. Организация работы по погашению просроченных ссуд.
5. Подготовка и направление документов в правоохранительные органы, если из материалов усматриваются признаки преступления.

Переговоры с потенциальным ссудозаемщиком сотрудники кредитного, юридического подразделений и подразделения безопасности могут вести как вместе, так и порознь. К достоинствам первого метода относится то, что сотрудники кредитной организации могут обменяться затем впечатлениями о клиенте, помогать друг другу в процессе беседы. Кроме того, то, что не заметит один из них в настораживающем поведении клиента, может увидеть другой. Да и клиенту, если он потенциальный мошенник, легче обмануть одного сотрудника кредитной организации, нежели сразу троих. С другой стороны, когда переговоры ведутся каждым сотрудником самостоятельно, можно более детально разобраться в специфических для данного сотрудника вопросах. В этом случае начинать их предпочтительнее сотруднику подразделения безопасности.

Эти три подразделения должны работать в тесном контакте, помогать и консультировать друг друга. Такая совместная работа должна начинаться уже в момент переговоров с потенциальным клиентом. В ходе переговоров каждый сотрудник выясняет сведения о будущем заемщике, исходя из своих функциональных обязанностей.

Предложенный вариант переговоров с потенциальным ссудозаемщиком принципиально отличается от существующего в практике многих кредитных организаций. Эти переговоры в подавляющем большинстве ведет кредитный инспектор. Участие же в них юристов и сотрудников подразделения безопасности дает возможность выяснить специфические для них вопросы, обратить внимание на те или иные моменты, предусмотреть возможные меры воздействия к клиентам, вовремя не возвратившим ссуды.

Исходя из решения задач, стоящих перед каждым подразделением, их сотрудники требуют от будущего заемщика необходимые им документы. В ряде случаев одни и те же документы необходимы сотрудникам разных подразделений, но используют они их с различных позиций, для решения своих задач.

Очень важное значение в работе кредитной организации и, прежде всего, кредитного подразделения и подразделения безопасности должно быть уделено контролю (мониторингу) за выданным кредитом, т.е. выявлению фактов, которые должны ее «насторожить». К их числу относятся следующие основные факты:

- резкое отклонение от условий использования кредита со ссылкой на незначительные факторы, не оказывающие существенного влияния на его исполнение, либо несоответствующие действительности факты;
- длительная задержка с началом исполнения кредитуемой сделки, трудно объяснимое поведение руководителей, в том числе уклонение по различным причинам от личных встреч, телефонных переговоров, оставление без ответа направляемых телеграмм; систематическое направление извинений о временных финансовых трудностях; постоянное требование о пролонгации кредита; неуплата налогов, таможенных платежей и стремление скрыть такие факты от кредитной организации;
- создание препятствий при проверке залога, договоров по кредитуемой сделке, бухгалтерской отчетности и других данных;
- заключение нелогичных с хозяйственной точки зрения, практически невыполнимых договоров;
- наличие конфликтных ситуаций на фирме заемщика, серьезные кадровые изменения, а также радикальные изменения в составе учредителей, акционеров, админи-

страции предприятия; существенные сокращения численности работников предприятия;

- резкое изменение профиля деятельности предприятия; потеря важных партнеров и клиентов; распродажа имущества, разрыв или непродление договора аренды помещений;
- выезд руководителей фирмы и членов их семей за границу; упорное распространение сведений о несостоятельности либо получение данных об умышленном создании неплатежеспособности и др.;
- несоразмерная зарплата, приобретение в личное пользование руководящими лицами дорогих вещей и недвижимости; приобретение ими недвижимости за границей и т.п.;
- получение сведений о злоупотреблениях руководящих лиц; совершение противоправных операций с денежными и материальными ценностями предприятия; участие в легализации преступных доходов; получение данных о связях с криминальными элементами и преступными группировками; введение в состав учредителей авторитетов преступного мира или криминальных групп либо назначение их на руководящие должности [5].

В настоящее время актуальной является неоднократно высказываемая в кругах подразделений безопасности банковских учреждений идея о создании единой базы данных о недобросовестных заемщиках, к которой имели бы доступ все кредитные организации. Такая система явилась бы серьезной преградой на пути преступников, которые в принципе не смогли бы получить и не возвращать десятки кредитов в различных кредитных организациях, пользуясь их неинформированностью. Решение данного вопроса всецело зависит от общего решения руководителей кредитных организаций. Так, в целях снижения кредитного риска в Государственной Думе РФ обсуждается соответствующий законопроект с предложением об учреждении специализированного Федерального государственного архива кредитных историй.

В результате проведенного анализа рассмотрена проблема защиты банковских интересов от преступных посягательств со стороны недобросовестных заемщиков. Сделаны выводы о необходимости создания единой базы данных о недобросовестных заемщиках, к которой имели бы доступ все кредитные организации.

## Литература


1. Уголовный кодекс Российской Федерации.
2. Калугин Н. Организация борьбы с мошенничеством в сфере банковского кредитования // Информационно-аналитический бюллетень «Банкир» Главного управления Банка России по Санкт-Петербургу. 2002. № 2.
3. Амплеев С.В. Меры защиты банков от незаконного получения кредитов. / Материалы учебно-практической конференции «Актуальные проблемы безопасности банковского дела в условиях финансового кризиса». М., 1999.
4. Чупрова А.Ю. Квалификация преступного обмана в кредитно-финансовой сфере. // Банковское право. 2000. № 2, С. 64–67.
5. Ларичев В. Преступления, совершаемые в сфере банковского кредитования. // Деньги и кредит. 1998. № 4. С. 70–72.
6. Ларичев В. Предупреждение работниками банка мошенничества и иных злоупотреблений, связанных с выдачей ссуд. // Деньги и кредит. 1997. № 3. С. 44–47.

# МОДЕЛИРОВАНИЕ СИСТЕМЫ БЕЗОПАСНОСТИ CMS В УСЛОВИЯХ ОГРАНИЧЕННОГО БЮДЖЕТА

**Е.В. Симаков, В.В. Заря, А.А. Протченков**
**Научный руководитель – д.т.н., профессор А.Г. Коробейников**

В статье описывается метод моделирования системы безопасности CMS, учитывающий существующие угрозы для CMS, степень ущерба от преодоления системы защиты, а также трудности при эксплуатации системы защиты. Описываемая методика призвана обеспечить минимальную трудоемкость и финансовые затраты при построении системы безопасности CMS.

## Введение

CMS (CMS – content management systems) или система управления контентом – это программное обеспечение, с помощью которой любой пользователь может вносить изменения на сайт так же, как в текстовом редакторе: с легкостью добавлять разделы, размещать иллюстрации, управлять рассылками, рекламными кампаниями, публиковать закрытую информацию, доступ к которой есть только у определенных групп пользователей. При этом подразумевается, что от пользователей такой системы не требуется специальных знаний технологий, отличающихся от обычно используемых в офисных процессах (текстовый редактор, Интернет и т.п.). При этом не следует считать, что такая система не требует обучения персонала, но это обучение касается порядка работы в системе, а не изучение новых технологий. И это только небольшой список всего того, чего можно добиться с помощью CMS.

В общем виде работу сайта на базе CMS можно представить в следующем виде. В системе присутствует два хранилища. В первом (обычно это реляционная СУБД) хранятся все данные, которые публикуются на сайте. Во втором (обычно это файловая система) хранятся элементы представления – шаблоны, графические изображения и т.д.

Кроме внешнего представления сайта, каким его видят все пользователи, есть как минимум два специализированных рабочих места. Первое рабочее место – для разработчиков сайта. С его помощью они задают структуру сайта, структуру контента, определяют внешний вид сайта, настраивают шаблоны представления информации. Этот инструментарий обычно не полностью автоматизирован. Для настройки сайта разработчики частично работают через средства CMS, часть информации размещается напрямую. Второе рабочее место – для владельцев сайта. Оно позволяет сотрудникам компании самостоятельно размещать информацию на сайте, без участия разработчиков. Менеджеры заказчика работают только через специализированное рабочее место.

Использование CMS предоставляет следующие очевидные преимущества.

- Оперативное обновление информации – информацию публикует сотрудник, владеющий информацией, без дополнительных посредников в виде технических специалистов.
- Снижение стоимости поддержки – обновление информация производится самостоятельно, нет необходимости оплачивать труд собственного или внешнего веб-мастера.
- Предоставление дополнительных сервисов пользователю – часть сервисов, таких как поиск, форумы, голосования и т.д., требуют интерактивного взаимодействия с пользователем. Они уже реализованы в рамках CMS.
- Уменьшение сроков и стоимости разработки – наиболее востребованная функциональность уже реализована в CMS и может быть сразу использована.
- Повышение качества разработки – при разработке полностью или частично используются готовые модули, которые уже прошли неоднократное тестирование.
- Снижение стоимости дальнейших модификаций – CMS позволяют разделить данные и их представление. Это позволяет гораздо проще изменить внешний вид сайта, чем в случае со статическим сайтом.

При построении системы безопасности CMS встает два непростых вопроса: во-первых – сколько средств необходимо потратить на построение системы безопасности, а во-вторых – какие средства защиты следует использовать, чтобы максимально обеспечить безопасность системы CMS, не выходя при этом за рамки установленного бюджета. Для обоснования стоимости системы защиты существует как минимум два подхода.

Первый подход, который можно назвать наукообразным, заключается в том, чтобы освоить, а затем и применить на практике необходимый инструментарий получения метрики и меры безопасности, а для этого привлечь собственника информации к оценке стоимости защищаемой информации, определению вероятностей потенциальных угроз и уязвимостей, а также оценке потенциального ущерба. В таком случае от результатов таких оценок будет во многом зависеть дальнейшая деятельность в области информационной безопасности. Если информация не стоит ничего, существенных угроз для информационных активов нет, а потенциальный ущерб минимален, и клиент это подтверждает, проблемой информационной безопасности можно, наверно, не заниматься. Если информация стоит определенных денег, угрозы и потенциальный ущерб ясны, то понятны и рамки бюджета на систему информационной безопасности. Существенно, что при этом становится возможным вовлечь клиента и заручиться его поддержкой в осознании проблем информационной безопасности и построении системы защиты информации.

Второй подход можно назвать практическим, и состоит он в следующем: можно попробовать найти инвариант разумной стоимости системы защиты информации. Информационной безопасностью CMS можно вообще не заниматься, и не исключен такой вариант, что принятый риск себя вполне оправдает. А можно потратить на создание системы защиты немало денег, и все равно останется некоторая уязвимость, которая рано или поздно приведет к утечке или хищению конфиденциальной информации.

Второй вопрос – выбор набора контрмер, которые будут использоваться для защиты CMS, – является еще более сложным. Ведь каждый клиент и каждая система CMS должны иметь индивидуальный подход в решении этого вопроса. И ответы на этот вопрос для разных клиентов будут отличаться друг от друга. При этом ответ формируется под действием нескольких факторов, таких как: область применения конкретной CMS, стоимость информации, обрабатываемой в системе, угрозы, направленные против конкретного сайта, вероятность проявления этих угроз, ущерб, наносимый владельцу информации при реализации данных угроз, средств, которые клиент готов потратить на организацию безопасности и большого набора существующих средств защиты (контрмер). Учесть такое разнообразие факторов, влияющих на будущий облик системы защиты, можно при проектировании одного-двух сайтов, но если проектирование и изготовление сайтов является основным видом деятельности компании, которая выпускает сайты десятками или даже сотнями, то на исследования и проектирование системы безопасности каждый раз будет уходить большое количество средств и времени. А это, в свою очередь, вызовет увеличение стоимости всего проекта и сроков его реализации. Рассматриваемая в данной статье методика моделирования системы безопасности для CMS направлена на решение поставленных проблем.

## Описание разработки и работы предлагаемой методики

При разработке методики моделирования системы обеспечения безопасности CMS необходимо учесть несколько основных (ключевых) факторов. Во-первых, необходимо определить рамки бюджета обеспечения информационной безопасности. Тут можно выделить два подхода. Первый – если клиент заранее знает, какую сумму он готов потратить на защиту своего сайта. Существенно, что при этом становится возможным вовлечь и заручиться поддержкой клиента, так как в данном случае подразумевается, что он осознает проблемы построения системы защиты информации. Второй под-

ход можно назвать практическим, и состоит он в нахождении инварианта разумной стоимости корпоративной системы защиты информации. Эксперты практики в области защиты информации пришли к выводу, что стоимость системы информационной безопасности не должна превышать 10–15 % стоимости CMS в зависимости от критичности информации, циркулирующей в ней. Это и есть та самая оценка (best practice), которой можно уверенно оперировать.

Поэтому первым шагом можно считать определение затрат на информационную безопасность (затраты на информационную безопасность будем обозначать в дальнейшем $SC_1$ – Security Cost). Если клиент затрудняется с определением стоимости системы безопасности, то определить данные затраты можно по формуле (рис. 1)

$$SC_1 = 0.15 \times TC, \quad (1)$$

где $TC$ – стоимость информационного ресурса ($TC$ – Total Cost).

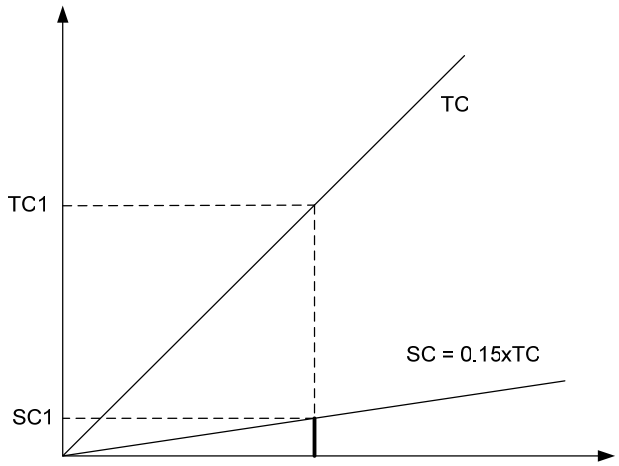



Рис. 1. Определение затрат на информационную безопасность

На втором этапе необходимо определить угрозы и вероятности их проявления для данного сайта. Определение угроз и коэффициентов их проявления также можно провести двумя способами. Первый способ является более общим и предназначен для клиентов, которые затрудняются определить вероятность проявления тех или иных угроз. Для такого класса клиентов коэффициенты угроз определяются заранее с помощью экспертной оценки для различных типов сайтов, которая обобщенно представлена в табл. 1. Если клиент затрудняется с определением коэффициентов угроз, то в таком случае необходимо определить класс сайта и использовать стандартные значения из таблицы. Такой подход обеспечивает более грубое определение коэффициентов угроз и является менее эффективным, но в то же время и менее трудоемким, чем второй подход.

| | сайты: визитка, витрина | сайты Интернет-магазинов | сайты корпоративных представительств | информационно-новостные сайты | сайты гос. структур |
|---|---|---|---|---|---|
| Хищение | $A_1$ | $A_2$ | $A_3$ | $A_4$ | $A_5$ |
| Утрата | $B_1$ | $B_2$ | $B_3$ | $B_4$ | $B_5$ |
| Блокирование | $C_1$ | $C_2$ | $C_3$ | $C_4$ | $C_5$ |
| Уничтожение | $D_1$ | $D_2$ | $D_3$ | $D_4$ | $D_5$ |
| Модификация | $E_1$ | $E_2$ | $E_3$ | $E_4$ | $E_5$ |
| Навязывание ложной информации | $F_1$ | $F_2$ | $F_3$ | $F_4$ | $F_5$ |

Таблица 1. Коэффициенты угроз для различных типов сайтов

Во втором подходе клиент самостоятельно оценивает угрозы, которые, по его мнению, будут наиболее критичными для создаваемого сайта. Клиенту предлагается определить коэффициенты угроз от 0 до 100 для следующих угроз: хищение, утрата, блокирование, уничтожение, модификация и навязывание ложной информации. В таком случае результаты моделирования системы безопасности будут наиболее приближены к ожиданиям клиента. Но данный подход требует от заказчика полного понимания того, кем и как будет использоваться информационный ресурс и как будут распределены угрозы информационной безопасности.

Независимо от того, какой из подходов выберет для себя клиент, после прохождения данного шага получается набор коэффициентов угроз ($t_j$ – threat):

$t_1$ – коэффициент угрозы хищения;

$t_2$ – коэффициент угрозы утраты;

$t_3$ – коэффициент угрозы блокирования;

$t_4$ – коэффициент угрозы уничтожения;

$t_5$ – коэффициент угрозы модификации;

$t_6$ – коэффициент угрозы навязывания ложной информации.

Чтобы можно было определить эффективность каждой контрмеры и ранжировать их, требуется проведение экспертной оценки. Так как каждая контрмера имеет свою эффективность по решению (противодействию) каждой угрозы, то для этого выделяются коэффициенты эффективности («веса») для каждой контрмеры по решению каждой угрозы. Веса ранжируются в диапазоне от 0 до 1. Чем больше коэффициент, тем более эффективно использование данного средства защиты по снижению вероятности угрозы. Также для каждой из контрмер требуется определение оценочной стоимости разработки и эксплуатации каждой контрмеры ($c_i$ – cost – стоимость). Полученные данные в общем виде показаны в табл. 2.

| Контрмеры (U) | Хищение | Утрата | Блок. | Унич. | Модиф. | Навяз. | Cost |
|---|---|---|---|---|---|---|---|
| 1. Подсистема идентификации и управления доступом | | | | | | | |
| $U_1$ | $V_{1x1}$ | $V_{1x2}$ | $V_{1x3}$ | $V_{1x4}$ | $V_{1x5}$ | $V_{1x6}$ | $C_1$ |
| $U_2$ | $V_{2x1}$ | $V_{2x2}$ | $V_{2x3}$ | $V_{2x4}$ | $V_{2x5}$ | $V_{2x6}$ | $C_2$ |
| 2. Подсистема регистрации и учета | | | | | | | |
| $U_3$ | $V_{3x1}$ | | | | | $V_{3x6}$ | $C_3$ |
| $U_4$ | … | … | … | … | … | … | … |
| 3. Криптографическая подсистема | | | | | | | |
| … | … | … | … | … | … | … | … |
| … | … | … | … | … | … | … | … |
| 4. Подсистема обеспечения целостности | | | | | | | |
| … | … | … | … | … | … | … | … |
| $U_n$ | $V_{nx1}$ | $V_{nx2}$ | $V_{nx3}$ | $V_{nx4}$ | $V_{nx5}$ | $V_{nx6}$ | $C_n$ |

Таблица 2. Коэффициенты эффективности контрмер

В этой таблице представлены контрмеры ($u_i$), сгруппированные по соответствующим подсистемам. Каждая из контрмер имеет свою стоимость $c_i$, независимо от угрозы, против которой направлена данная контрмера. Стоимость данного средства защиты остается неизменным, сколько бы угроз ни устраняла данная контрмера. Также каждая контрмера имеет свой коэффициент эффективности решения (свой «вес») – $v_{ixj}$. Но для решения поставленной задачи необходимо ответить на вопрос: насколько каждая контрмера эффективна в обеспечении безопасности всего сайта, а не каждой угрозы в отдельности. Чтобы ответить на этот вопрос, нужно определить коэффициент эффек-

тивности использования данной контрмеры для решения конкретной угрозы. Для этого необходимо в рамках всей таблицы провести следующие вычисления:

$$u_i\ (t_j \times v_{ixj};\ c_i). \qquad (2)$$

В результате получается коэффициент, который отражает, насколько использование каждой контрмеры оправдано для решения данной угрозы. Если какая-то угроза является для незначительной, то и результирующий коэффициент окажется небольшим, а это будет означать, что использование данной контрмеры для устранения данной угрозы не является приоритетным. Такой же результат будет и в ситуации, когда ощутимая угроза будет парироваться контрмерой, эффективность которой невелика. В результате получится небольшой результирующий коэффициент. Совершенно другая картина будет наблюдаться в случае, когда значимая угроза будет решаться эффективным средством защиты. Получится большой результирующий коэффициент, и это покажет, что при дальнейшем моделировании системы обеспечения безопасности необходимо будет сначала обратить внимание именно на данное средство защиты.

После преобразования получаются результаты, которые можно представить в табл. 3, характеризующей эффективность использования тех или иных контрмер для решения угроз безопасности информации, в зависимости от значений коэффициентов угроз. Получаются значения, которые будут отражать, насколько эффективно использование тех или иных контрмер для снижения тех или иных угроз.

| $t_1$ | $t_2$ | $t_i$ | $t_6$ |
|---|---|---|---|
| $u_1\ (t_1 \times v_{1x1};\ c_1)$ | $u_1\ (t_2 \times v_{2x1};\ c_1)$ | … | $u_1\ (t_6 \times v_{6x1};\ c_1)$ |
| $u_2\ (t_1 \times v_{1x2};\ c_2)$ | $u_2\ (t_2 \times v_{2x2};\ c_2)$ | … | $u_1\ (t_6 \times v_{6x2};\ c_2)$ |
| … | … | … | … |
| $u_i\ (t_1 \times v_{1xi};\ c_i)$ | $u_i\ (t_2 \times v_{2xi};\ c_i)$ | … | $u_i\ (t_6 \times v_{6xi};\ c_i)$ |

Таблица 3. Эффективность использования контрмеры для ликвидации угроз

На следующем шаге требуется определить эффективность каждой контрмеры в рамках всего сайта. Для этого необходимо преобразовать таблицу таким образом, чтобы узнать, насколько использование каждой контрмеры оправдано для данной CMS. Чтобы определить коэффициент такой эффективности, необходимо сложить эффективность каждой контрмеры в каждой строке таблицы. Так как в каждой строке содержатся значения эффективности по решению отдельных угроз, то после их суммирования получится значение, которое будет отражением того, насколько использование данной контрмеры эффективно в рамках всей CMS.

Обозначив $t_i \times v_{i\,x\,i}, = p_i$ (p – profit), получим:

$$\begin{aligned} u_1 &= (p_{1x1} + p_{1x2} + \ldots + p_{1x6};\ c_1);\\ u_2 &= (p_{2x1} + p_{2x2} + \ldots + p_{2x6};\ c_2); \qquad (3)\\ &\ldots\\ u_i &= (p_{i\,x\,1} + p_{i\,x\,2} + \ldots + p_{i\,x\,6};\ c_i). \end{aligned}$$

Если обозначить

$$p_i = p_{i\,x\,1} + p_{i\,x\,2} + \ldots + p_{i\,x\,6}, \qquad (4)$$

то для каждой контрмеры получатся следующие значения:

$$\begin{aligned} u_1 &= (p_1;\ c_1);\\ u_2 &= (p_2;\ c_2); \qquad (5)\\ &\ldots\\ u_i &= (p_i;\ c_i). \end{aligned}$$

Следовательно, чем больше значение $p_i$, тем эффективнее использование данной контрмеры в пределах всей системы CMS.

Теперь, когда после всех преобразований получен определенный набор контрмер с определенными весами и стоимостью, встает главная задача – оптимальный выбор: из

всех контрмер, которые можно использовать, выбрать такой набор, который при заданном ограничении по стоимости (затраты не должны превышать значения *SC*, полученного ранее) будет иметь наибольшую эффективность, т.е. максимальное суммарное значение $p_i$. Данную задачу можно представить в следующем виде:

$$\max S = \sum_{i=1}^{n} p_i \qquad (6)$$

при ограничении

$$\sum_{i=1}^{n} c_i \le SC . \qquad (7)$$

Для решения задачи можно воспользоваться «жадным алгоритмом» – найти относительную ценность каждой контрмеры на единицу стоимости и при выборе контрмер пользоваться уже значением именно этого параметра. Но при тестировании данного алгоритма был выявлен один очень существенный недостаток: он не обеспечивает требуемой точности, а иногда в результате его работы полученный набор контрмер не являлся оптимальным.

Чтобы гарантированно получить оптимальное решение данной задачи, надо воспользоваться методом декомпозиции – использовать такой алгоритм, в котором была бы организована рекурсия для расширения «хорошего» частного решения. В данной методике используются методы динамического программирования для реализации такой задачи. Из заданного набора контрмер путем последовательного перебора необходимо отбирать такие решения, которые дают увеличение ценности данного набора контрмер. Алгоритм решения этой задачи представлен на рис. 2.

Для решения задачи используются 2 массива: двумерный (P[i,j]) – для поиска максимально хорошего решения, одномерный (S[(i)(j)]) – для хранения идентификаторов тех контрмер, которые вошли в каждое частное решение. Строки двумерного массива (i) соответствуют используемым контрмерам, количество столбцов (j) соответствует значению *SC*, а в каждую ячейку i×j заносятся частные значения ценности набора контрмер i при общей стоимости набора j. Как уже говорилось выше, массив P[i,j] используется для перебора всех возможных решений, при этом для каждой контрмеры происходит сравнение: что лучше использовать при данной стоимости – просто добавить в набор рассматриваемую контрмеру или удалить из набора определенную контрмеру, взятую в предыдущем шаге. При этом каждой ячейке массива P[i,j] в соответствие поставлена ячейка одномерного массива S[(i)(j)] в которую заносятся идентификаторы контрмер, с использованием которых была получена ценность набора в ячейке P[i,j].

1. На первом этапе инициализируются используемые массивы. Причем проинициализировать массив P[i,j] необходимо в пределах i є [0,n] (где n – количество контрмер) и j є [0,SC] (где SC – максимальные затраты на обеспечение безопасности CMS).
2. Переменной i присваивается значение равное 1 (т.е. используется первая контрмера).
3. Переменной j присваивается значение 1.
4. Проверяется, не превышает ли стоимость данной контрмеры значения j.
5. Если не превышает, то тогда перейти к шагу 6, если превышает – перейти к шагу 7.
6. Следует сравнить, какой из наборов контрмер представляет большую ценность. Если тот, который был получен в предыдущем проходе при использовании контрмер от 1 до i-1 для данного ограничения по стоимости (j), то перейти к шагу 7. Если же будет эффективнее использовать данную контрмеру плюс набор контрмер, стоимость которых в совокупности со стоимостью данной контрмеры не будет превышать значения j, то тогда перейти к шагу 10.
7. Присвоить значению P[i,j] значение, равное P[i-1,j], значению S[(i)(j)] – значение, равное S[(i-1)(j)].
8. Увеличить значение j на единицу.
9. Если j меньше или равно SC то перейти к шагу 4, если нет – то к шагу 11.

10. Присвоить значению P[i,j] значение равное p[i] + P[i-1,j-c[i]], а в массив S[ij] занести значение S[(i-1)(j-c[i])] и прибавить к нему идентификатор используемой контрмеры. Перейти к шагу 8.
11. Увеличить идентификатор контрмеры (i) на единицу.
12. Если значение i меньше или равно n (то есть пока все контрмеры не использованы) – перейти к шагу 3, если все контрмеры уже были использованы то перейти к шагу 13.
13. Вывести значение ячейки S[(i)(j)].

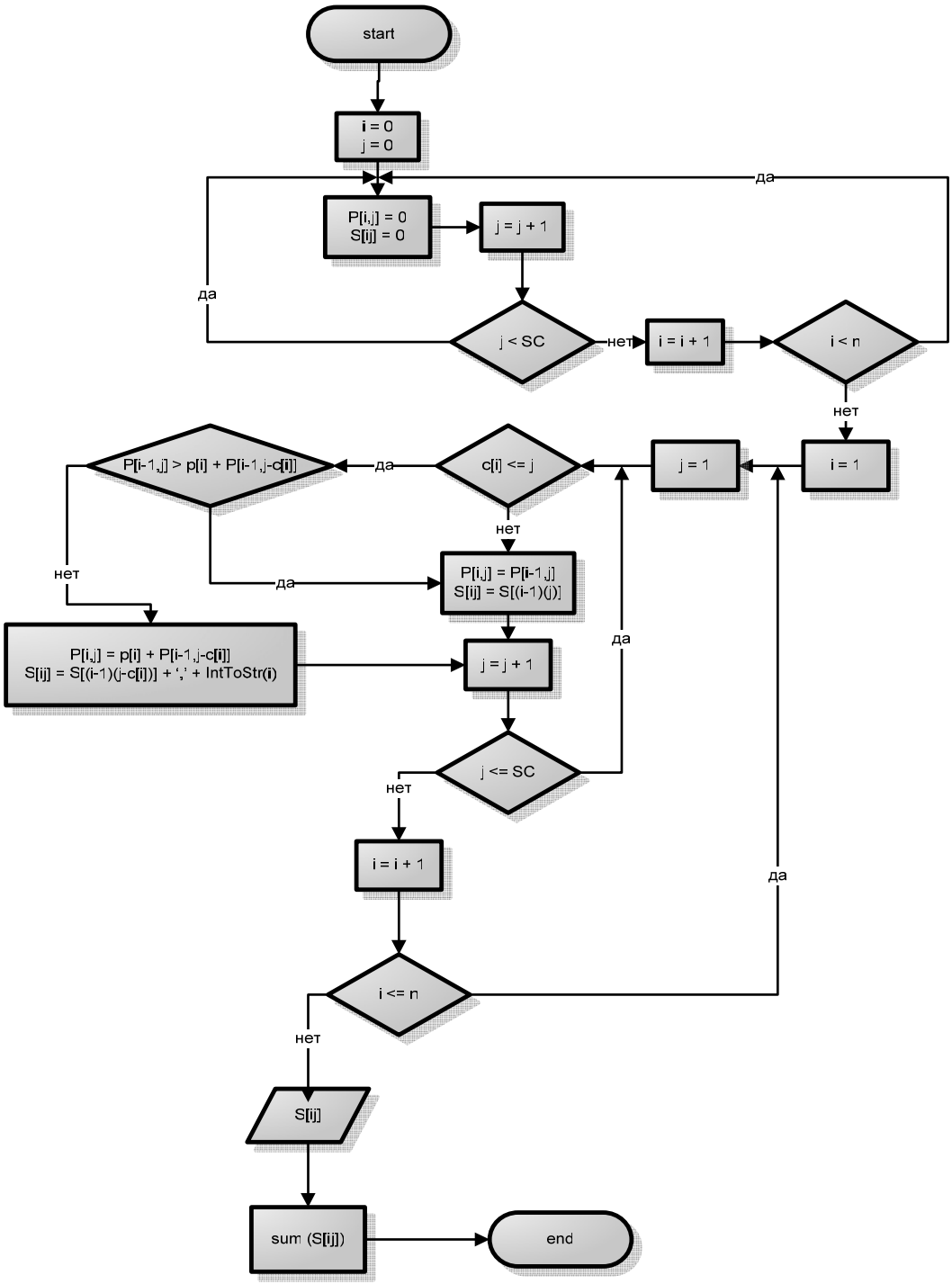



Рис. 2. Алгоритм оптимального выбора

Таким образом, в ячейке S[(i)(j)] будет записан оптимальный набор контрмер, который удовлетворяет условиям поставленной задачи. Ценность данного набора будет максимальна (в чем можно убедиться, сравнив значение ячейки P[i,j] с другими ячей-

ками двумерного массива P). При этом стоимость набора контрмер можно определить, если сложить стоимости всех контрмер, вошедших в набор, и суммарная стоимость полученного набора не будет превышать выделенных средств на обеспечение безопасности CMS.

## Заключение

Описываемая методика позволяет решить два главных вопроса, встающих при создании системы безопасности CMS: первый – сколько средств необходимо потратить на построение системы безопасности, и второй – какие средства защиты следует использовать, чтобы максимально обеспечить безопасность системы CMS, не выходя при этом за рамки установленного бюджета.

К достоинствам предлагаемой методики можно отнести то, что она:

- учитывает ключевые факторы, обусловливающие создание системы безопасности CMS;
- учитывает финансовые затраты на создание системы безопасности;
- учитывает степень ущерба от преодоления системы безопасности CMS;
- учитывает тип защищаемого сайта и вероятности угроз для каждого из них;
- учитывает эффективность средств защиты CMS от существующих угроз безопасности;
- обеспечивает максимальную эффективность системы безопасности в рамках заданного бюджета и условий эксплуатации сайта;
- обеспечивает минимальную трудоемкость и финансовые затраты при работе с методикой.

## Литература


1. Петренко С.А., Симонов С.В. Экономически оправданная безопасность. М: ДМК Пресс, 2004. 378 с.
2. Петренко С.А., Петренко А.А. Аудит безопасности Intranet. М: ДМК Пресс, 2002. 416 с.
3. Русел Р., Бидвел Т. Защита сайта электронной торговли. Syngress Publishing Inc. 2001. 690 с.
4. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: построение и анализ. М: МЦНМО, 2000. 960 с.

# АНАЛИЗ И БЕЗОПАСНОСТЬ СЕТЕВОГО СТЕКА ОС WINDOWS VISTA

**Д.С. Туранцев**
**Научный руководитель – д.т.н., профессор Ю.А. Гатчин**

Статья является техническим обзором сетевых и коммуникационных усовершенствований в ОС Windows Vista для решения вопросов связи, удобства использования, управления, повышения надежности и обеспечения безопасности.

## Введение

В ОС Windows Vista реализована новая версия стека TCP/IP, существенным образом улучшающая несколько наиболее важных аспектов сетевой работы и позволяющая добиться повышения производительности и пропускной способности, а также собственная архитектура Wi-Fi и интерфейсы API для проверки сетевых пакетов.

Для максимального использования сетевых возможностей необходима комплексная настройка конфигурационных параметров TCP/IP. В Windows Vista не приходится делать это вручную, так как система сама анализирует сетевые условия и автоматически оптимизирует сетевые параметры. В сетях с большими потерями данных, например, в беспроводных сетях, Windows Vista способна лучше восстанавливать информацию после потери одного или нескольких пакетов. Она может динамически увеличивать или уменьшать окно TCP на прием, что позволяет использовать всю ширину канала. При передаче файлов по высокоскоростной глобальной сети с большим временем отклика или при скачивании файлов из сети Интернет пользователи, несомненно, заметят существенное сокращение времени передачи файлов.

Кроме того, в состав базового сетевого стека Windows Vista включена собственная архитектура беспроводного соединения (собственный интерфейс Wi-Fi). К числу его преимуществ можно отнести гибкое использование во многих моделях устройств различных торговых марок, схожие приемы работы с разными устройствами и более надежные драйверы беспроводных сетевых карт независимых поставщиков. Беспроводными сетями в ОС Windows Vista можно управлять централизованно, причем соединения по таким сетям поддерживают новейшие протоколы безопасности и позволяют пользователям работать с меньшими задержками.

В стеке TCP/IP нового поколения реализована новая архитектура сетевой защиты Windows Filtering Platform (WFP) с интерфейсами API, позволяющими независимым разработчикам программного обеспечения участвовать в процессе принятия решений о фильтрации пакетов на нескольких уровнях стека протокола TCP/IP без написания собственных приложений привилегированного режима. Эта архитектура обеспечивает поддержку таких функций сетевого экрана нового поколения, как проверка подлинности при соединении и динамическое конфигурирование сетевого экрана при использовании приложениями интерфейса Windows Sockets API (политика, зависящая от приложения).

## Облегчение задач пользователей

Узнать о состоянии сети, т.е. проверить наличие соединения, выяснить провайдера соединения, узнать, в местной сети или в сети Интернет они находятся, пользователи теперь могут из единого центра управления сетевыми возможностями (см. рис. 1). Кроме того, они могут просматривать состояние различных сетевых служб на своих компьютерах. Виден ли компьютер в местной сети? Какие папки и принтеры открыты у них для доступа по сети? Пользователь может создать сеть (временную или инфра-

структурную беспроводную сеть, сеть VPN или домашнее широкополосное соединение) или подключиться к существующей сети любого типа.

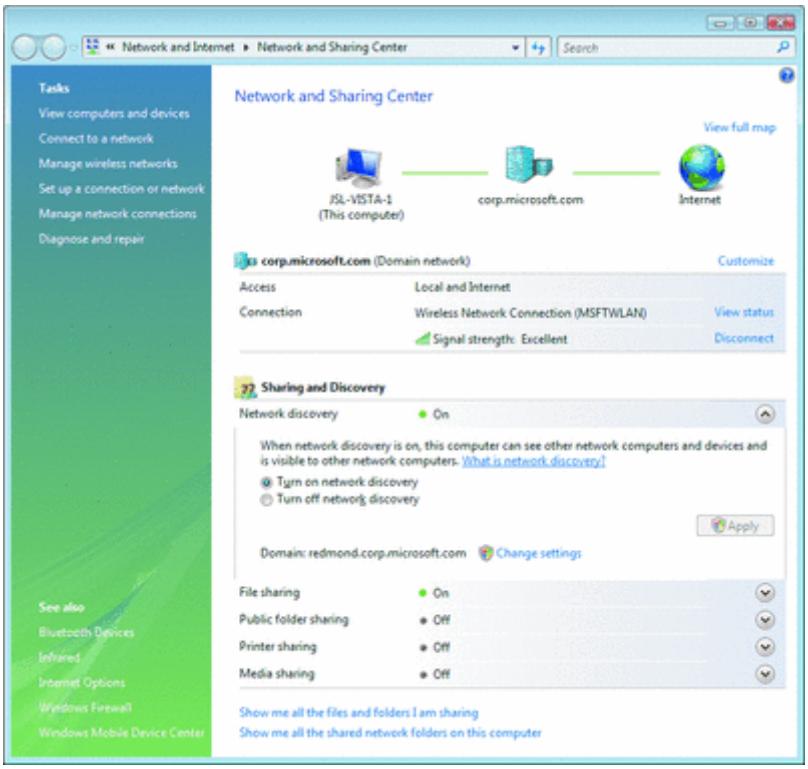



Рис. 1. Центр управления сетевыми возможностями

Система Windows Vista способна самостоятельно диагностировать и разрешать многие проблемы со связью, так что пользователю не приходится обращаться в службу техподдержки. Инфраструктура диагностики сетевого соединения (Network Diagnostics Framework) позволяет ОС Windows Vista выявлять основные причины проблем со связью в контексте операции приложения. Например, если пользователь не может попасть на какой-либо Интернет-сайт, то данная система диагностики попытается отследить проблему по всей цепочке связи, начиная от определения наличия активного беспроводного соединения и действительного IP-адреса, вплоть до установления связи с DNS сервером, нахождения прокси-сервера и получения ответа от требуемого веб-сервера.

В случае определения причины проблемы пользователь получает сообщение с четким описанием проблемы и способов ее разрешения. Иногда проблема устраняется простым щелчком мыши на данном сообщении. В некоторых случаях пользователю придется внести изменения в настройки, и диалоговое окно доставит пользователя в необходимое место. А в случаях, когда пользователь просто не может выполнить необходимые действия по причине недостатка знаний или отсутствия прав, в обозреватель событий записываются более полные сведения, поэтому служба техподдержки может быстро устранить проблему, не тратя часы на поиск неполадок.

В ОС Windows Vista используются интерфейсы API Network Awareness, вызываемые приложениями для выяснения состояния соединения и определения типа сети (домен, сеть общего доступа или частная сеть), к которой подключен компьютер в настоящий момент. Если ОС Windows Vista может получить сетевой доступ к контроллеру домена, то она автоматически выбирает профиль «Домен». Другие сети в эту категорию попасть не могут. Все другие сети определяются как сети общего доступа, если пользователь или приложение не укажет, что сеть частная. Сети с прямым подключением к Интернет или сети в общественных местах, например, в аэропортах или Интернет-кафе, следует оставить в категории сетей общего доступа. К категории частных се-

тей следует относить только те сети, которые защищены частным межсетевым шлюзом, например, домашние сети или небольшие корпоративные сети.

Наличие интерфейса Network Awareness позволяет таким приложениям, как сетевой экран с дополнительными функциями безопасности (Windows Firewall with Advanced Security) (описываемый ниже), использовать различные настройки для сетей разного типа и переходить на эти настройки автоматически при изменении типа сети. Например, администратор может настроить сетевой экран таким образом, чтобы при подключении компьютера к сети с доменом определенные порты для программы управления рабочим столом были открыты, но автоматически закрывались при работе в общедоступных хот-спотах.

Групповая политика в ОС Windows Vista также определяется типом сетевого подключения: при подключении компьютера к сети с доменом система автоматически начинает обрабатывать новые настройки групповой политики, не ожидая следующего цикла обновления. Это означает, что ОС Windows Vista автоматически запрашивает новые настройки групповой политики при подключении компьютера к сети с доменом даже в том случае, когда она выходит из спящего режима. Это позволяет администраторам более оперативно вводить новые настройки безопасности, когда время играет существенную роль.

### Обеспечение безопасности сети

При работе по сети существует несколько типов опасностей – подключение к беспроводным сетям злоумышленников, имитирующим сети общего доступа; подключение зараженных ПК к корпоративной сети; попытка неуправляемых ресурсов получить доступ к закрытым для них ресурсам. Перечисленные опасности могут загрузить сетевого администратора на весь рабочий день и заставить беспокойно ворочаться всю ночь. ОС Windows Vista может облегчить борьбу со всеми этими опасностями благодаря дополнительным функциям сетевой защиты, простым в настройке и всеобъемлющим одновременно.

Собственная архитектура Wi-Fi в Windows Vista имеет широкую поддержку новейших протоколов безопасности, в том числе корпоративной и персональной версии протокола Wi-Fi Protected Access (WPA) 2, протокола PEAP-TLS и протокола PEAP-MS-CHAP v2 (защищенный наращиваемый протокол аутентификации с обеспечением безопасности на транспортном уровне и с протоколом взаимной аутентификации). Такая широкая поддержка обеспечивает возможность взаимодействия между Windows Vista и почти всеми беспроводными устройствами. Windows Vista анализирует характеристики беспроводной сетевой карты, что позволяет по умолчанию выбрать наиболее безопасный протокол при подключении к беспроводной сети или создании такой сети. При помощи платформы EAP-HOST ОС Windows Vista может поддерживать специализированные механизмы аутентификации, разработанные поставщиком беспроводных устройств или какой-либо организацией.

В ОС Windows Vista реализованы многочисленные усовершенствования клиентской части беспроводного соединения, позволяющие отражать ненаправленные беспроводные атаки. Такой клиент автоматически подключается только к сетям, указанным пользователем в списке разрешенных сетей, или подключается по прямому требованию пользователя. К временным сетям он автоматически не подключается. Кроме того, клиент выдает предупреждение, если пользователь собирается установить соединение с ненадежной сетью. Активный поиск разрешенных сетей клиент осуществляет по сокращенному списку и только по указанию пользователя, что усложняет злоумышленникам задачу определения названия сети, к которой пользователь пытается подключиться, и подмены ее своей сетью с тем же именем.

Собственный клиент беспроводного соединения ОС Windows Vista поддерживает функцию единого входа (SSO), осуществляющую аутентификацию пользователя в сети на уровне Layer 2 в необходимый момент времени с учетом настроек сетевой безопасности, причем вход в сеть и вход в систему Windows при этом взаимосвязаны. После создания профиля единого входа вход в сеть будет осуществляться раньше входа в систему Windows. Эта возможность позволяет выполнять такие операции, как обновление групповой политики, запуск скриптов регистрации, начальная загрузка по беспроводной сети, требующие подключения к сети прежде входа пользователя в систему.

Сетевой экран с дополнительными функциями безопасности обеспечивает новый уровень сетевой защиты в системе Windows с поддержкой фильтрации входящих и исходящих пакетов и функции повышения стойкости служб (Windows Service Hardening). Если сетевой экран обнаруживает, что поведение какой-либо службы Windows отклоняется от нормального, описанного в сетевых правилах системы повышения стойкости служб, то он блокирует эту службу. Данный сетевой экран поддерживает и функцию разрешенного обхода (Authenticated Bypass), позволяющую некоторым компьютерам после проверки их подлинности службой IPsec обходить правила сетевого экрана для выполнения таких задач, как удаленное управление.

Одно из наиболее существенных изменений в сетевом экране заключается в его объединении со службой IPSec. Раньше для создания многоуровневой совокупности правил сетевой безопасности администраторам приходилось полагаться на два отдельных инструмента – сетевой экран и средство применения протокола IPsec и управления им. В Windows Vista для защиты сети от несанкционированного доступа администраторы могут создавать простые правила сетевой безопасности, объединяющие правила сетевого экрана и правила IPSec. Благодаря такому объединению можно осуществлять сквозную передачу данных по сети после установления подлинности обменивающихся сторон с обеспечением расширяемого многоуровневого доступа к доверенным сетевым ресурсам и/ или защиты конфиденциальности и целостности данных.

Администратор может логически разделить корпоративную сеть на зоны, доступ в которые может быть предоставлен любому компьютеру (в том числе с правами гостя), или только компьютерам, прошедшим аутентификацию в домене (отделение домена). Кроме того, администратор может отделить некоторые серверы, доступ к которым следует предоставлять только определенной группе пользователей или компьютеров, например, сервер приложений отдела кадров с разрешением доступа только компьютерам отдела кадров (отделение сервера), как показано на рис. 2.

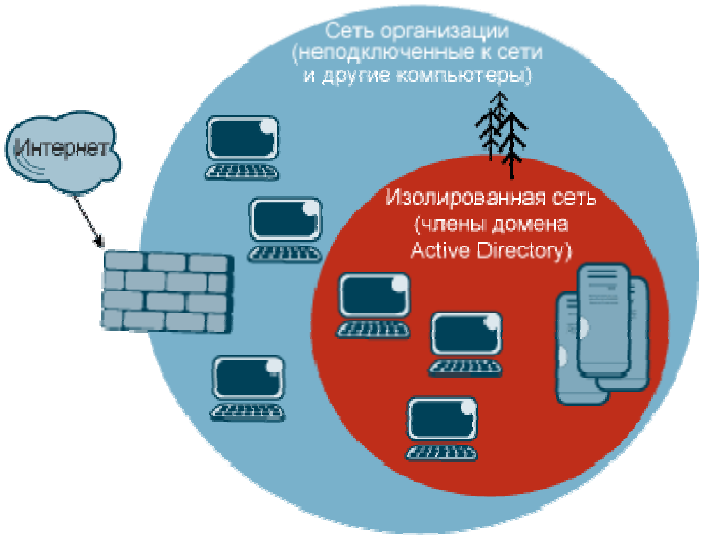



Рис. 2. Изоляция серверов и доменов

### Упрощение управления сетью

Сетевые возможности в ОС Windows Vista спроектированы с поддержкой управляемости по всем основным параметрам, позволяющей сократить расходы на внедрение беспроводных сетей и политик сетевой безопасности, а также обеспечить качество обслуживания приложений и пользователей. Для управления сетевыми функциями в ОС Windows Vista широко используются скрипты групповой политики или командной строки, выполняемые в сетевой оболочке NETSH, поэтому вам не требуется изучать или внедрять новый инструмент управления, а можно получить большую отдачу от вложенных средств в систему Active Directory® и использовать созданную вами структуру подразделения (OU).

Внедрение правил сетевой безопасности (с объединением политик сетевого экрана и службы IPsec) и управление этими правилами упрощается благодаря использованию одного встроенного в консоль управления MMC приложения (сетевого экрана с дополнительными функциями безопасности), сопровождающего пользователя подсказками, или скриптов командной строки, выполняемых в оболочке NETSH. Это новое встроенное приложение позволяет легко и удобно реализовать правила фильтрации входящих или исходящих потоков, а также правила установления безопасной связи, ограничивающие доступ конкретным пользователям, компьютерам или приложениям с обеспечением административного управления на уровне мельчайших деталей. Для обеспечения соответствия политике безопасности на основе сценариев служба IPSec может затребовать проверку подлинности пользователей, компьютеров или отсутствия вирусов (совместно с функцией защиты сетевого доступа). Встроенное приложение облегчает создание правил отделения сервера или домена, а поскольку оно работает на основе групповой политики, вы можете применять эти правила гибким образом, в зависимости от структуры вашей организации.

Оболочка NETSH позволяет автоматизировать поиск неполадок в беспроводных соединениях и использовать при этом скрипты. С командной строки администраторы могут проверять, изменять или удалять конфигурационные профили беспроводной сети. Эти конфигурационные профили можно экспортировать на другие компьютеры или импортировать из других компьютеров, что упрощает настройку нескольких компьютеров с одинаковыми функциями.

Вопреки всем заверениям Microsoft, сетевой стек ОС Windows Vista намного менее надежен и безопасен, чем в XP, к тому же он совершенно не изучен и абсолютно непредсказуем. У администраторов нет опыта решения проблем, с которыми они прежде не сталкивались, разработчики защитных компонентов (от программных брандмауэров до аппаратных комплексов) еще не включили поддержку ОС Windows Vista и ее протоколов в свои продукты.

Сетевой стек тесно интегрирован с ОС, и отделить его, вернув старый стек на место, никакой возможности нет. Нам предлагают множество новых компонентов, причем это предложение из разряда тех, от которого невозможно отказаться — ведь ни отключить, ни заблокировать ненужные функции все равно нельзя. То есть можно, конечно, но отнюдь не через графический интерфейс, и большинство пользователей этого сделать не сможет, а, значит, черви, хакеры и удаленные атаки будут процветать.

### Глубины безопасности сетевого стека ОС Windows Vista

Главным и, пожалуй, единственным достижением Microsoft'a стала интеграция IPv4 и IPv6 в единый стек (до этого они были реализованы как раздельные компоненты), что и плохо, и хорошо одновременно. Хорошо то, что конечный пользователь получает готовый IPv6 без всякой головой боли и установки дополнительных пакетов.

Катастрофическая нехватка IP-адресов с каждым сезоном ощущается все острее и острее, но переход на IPv6 сдерживается как необходимостью смены сетевого оборудования, так и обновлением серверных и клиентских ОС. В исторической перспективе переход на IPv6 неизбежен.

Большинству сегодняшних пользователей IPv6 совершенно не нужен, поскольку для локальной сети и IPv4 хватает с лихвой, а основная масса провайдеров еще не поддерживает IPv6 и в обозримом будущем переходить на него не собирается. Проблема в том, что IPv6 несет в себе множество нововведений, еще не обкатанных и не протестированных в планетарном масштабе.

Теоретически IPv6 обеспечивает более высокую производительность и лучшую защиту от атак, но практически вопросы производительности решаются «тонкой» настройкой опций TCP//IP-протокола, которые в Windows доступны лишь частично, и крайне отрывочно документированы в виде заметок в Knowledge Base.

Настойки по умолчанию стремятся удовлетворить сразу всех и каждого, в результате чего по-настоящему не удовлетворен никто. Отсутствие легальных рычагов управления не позволяет оценивать реальную производительность сетевого стека, и громкие заявления Microsoft'a, что в ОС Windows Vista стек намного более производителен, следует расценивать как пропаганду. Результаты тестов ни о чем не говорят! Тем более что в большинстве случаев реальный CPS определяется отнюдь не «качеством» сетевого стека, а загруженностью удаленного сервера, пропускной способностью каналов связи и так далее. Глупо ожидать, что, установив ОС Windows Vista на свой компьютер, мы «разгоним» свой модем хотя бы на десяток процентов...

Следствием интеграции IPv4 с IPv6 в единый сетевой стек стало появление туннельных протоколов Teredo, ISATAP, 6to4 и 6over4, причем Teredo уже успел попасть в RFC и осесть под номером 4380 [1].

Teredo инкапсулирует (упаковывает) IPv6-трафик внутрь IPv4-пакетов, используя протокол UDP, слабости и недостатки которого хорошо известны. Если два IPv6-узла разделены IPv4-сегментом сети (наиболее типичная на сегодняшний день конфигурация), ОС Windows Vista задействует Teredo, направляя запрос одному из публичных Teredo-серверов, который, в свою очередь, передает его получателю, фактически выполняя роль proxy-сервера.

Самое интересное, что Teredo позволяет обходить трансляторы сетевых адресов (они же NAT') и брандмауэры, причем это не ошибка, а его функция. Рассмотрим два узла, защищенные NAT, прямое взаимодействие между которыми невозможно. Но это оно по IPv4 невозможно, а если использовать Teredo-тоннель, то истинный адрес и порт назначения окажется скрыт в Teredo-заголовке, а в IPv4 попадает адрес узла, «смотрящего» в Интернет и UDP-порт самого Teredo (3544 порт), который, конечно, можно и закрыть, но как же тогда с остальными Vista-клиентами общаться?! Поскольку NAT не может установить ни реального целевого адреса, ни реального целевого порта протокола IPv6, он беспрепятственно пропускает IPv4-пакет. Это же самое относится и к другим защитным механизмам, не поддерживающим протокола Teredo.

Но это еще что! Поддержка новых протоколов – всего лишь вопрос времени. Переход на ОС Windows Vista означает переход на Teredo, а переход на Teredo навязывает глобальную маршрутизацию, заставляющую забыть о приватных IP-адресах, использующихся в локальных сетях и невидимых снаружи.

Плюс ко всему Виста поддерживает инкапсуляцию IPv4 в IPv4 и IPv6 в IPv6, что позволяет скрывать истинные целевые адреса и порты, вынуждая брандмауэры и другие защитные средства проводить скрупулезный анализ трафика, а это сразу же увеличивает потребности в памяти и мощности процессора со всеми вытекающими отсюда последствиями.

## TCP/IP и его свита

На самом деле, TCP/IP никогда не используются в «чистом» виде и всегда окружены свитой вспомогательных протоколов, причем далеко не все из них нужны домашнему пользователю. Привычка Microsoft пихать все в одну коробку без возможности отделить одно от другого дает о себе знать, и мы не можем удалить лишние протоколы, которые не только занимают системные ресурсы, но еще и служат источником потенциальных ошибок.

Сетевой стек ОС Windows Vista включает в себя следующие протоколы: ICMP; IGMP; IPV4; IPV6; ICMPV6, TCP; UDP; IP6; GRE; ESP; AH; 43; 44; 249; 251. Половина протоколов не нужна не только рабочим станциям, но и серверам, а многие из них даже не имеют собственного имени, ограничиваясь только номером. В частности, протоколы 43 и 44 отвечают за маршрутизацию и фрагментацию в IPv6. Причем, в ранних бетах посылка мусора по 43 протоколу вводила ОС Windows Vista в глубокую задумчивость, но через некоторое время она, как ни в чем не бывало, возвращалась к обработке сетевых запросов. А вот «сетевой мусор», переданный по 44 протоколу, обрушивал систему в голубой экран смерти. Сейчас это уже исправлено, но неизвестно, сколько еще ошибок реализации предстоит обнаружить.

Алгоритм сборки IP-пакетов изменился в худшую сторону, и частично перекрывающиеся пакеты теперь безжалостно отбраковываются как неверные, порочные и вообще недостойные существования (по всей видимости, программистам лень было топтать клавиатуру, вот они и «срезали углы»). Исключение составляет ситуация, когда два пакета перекрываются на 100 %, – тогда отбрасывается последний пакет в пользу первого, хотя LINUX-системы поступают с точностью до наоборот. Вообще же говоря, проблем со сборкой перекрывающихся пакетов у всех систем хватает, и каждая из них имеет свои особенности, в результате чего принятые данные искажаются до неузнаваемости или пакет не собирается вообще [2].

## Новые TCP-/UDP-порты

Нормальный клиентский узел вообще не должен содержать никаких открытых TCP-/UDP-портов! Он даже может не обрабатывать ICMP-сообщения, в частности, игнорировать echo-запросы (на чем основан ping) и не отправлять уведомлений о «казни» пакета с «просроченным» TTL (на чем основана работа утилиты tracert), хотя все это считается дурным тоном и создает больше проблем, чем их решает.

Наличие открытых портов указывает на присутствие серверных служб, обслуживающих удаленных клиентов. Каждая такая служба – потенциальный источник дыр, переполняющихся буферов и прочих лазеек, через которые просачиваются черви, а воинствующие хакеры берут компьютер на абордаж.

Вот неполный список портов, открываемых системой в конфигурации по умолчанию: IPV6 UDP; MS-RPC (135); NTP (123); SMB (445); ISAKMP (500); UPNP (1900); WEB SERVICES DISCOVERY (3702); WINDOWS COLLABORATION (54745); СОВМЕСТНЫЙ ДОСТУП К ФАЙЛАМ И ПРИНТЕРАМ (137, 138); ПОРТЫ-ПРИЗРАКИ - (49767, 62133); IPV4 UDP; TEREDO (4380, 61587); MS-RPC (135); SMB (445); NBT (139); NRP 3540; IPV4 TCP; P2P GROUPING MEETINGS (3587); WINDOWS COLLABORATION (54744); СОВМЕСТНЫЙ ДОСТУП К ФАЙЛАМ И ПРИНТЕРАМ (137, 138).

Изобилие открытых портов создает серьезную угрозу безопасности. IPv6 отображает часть UDP-портов на IPv4, однако забывает «объяснить» этот факт своему же собственному брандмауэру, и если мы закрываем печально известный 135-й порт на IPv4, его необходимо закрыть также и на IPv6, равно как и наоборот.

В ранних бетах факт закрытия портов было очень легко обнаружить, поскольку при попытке установки соединения с несуществующим портом система возвращала пакет с флагом RST (как, собственно, и положено делать по RFC). Соответственно, порты, не возвратившие пакета с таким флагом, но и не установившие соединения, все-таки существуют, но закрыты брандмауэром, который можно легко обойти, например, через RPC. Правда, эта лазейка была быстро закрыта, но зато при отправке сообщения на несуществующий UDP IPv6-порт до сих пор возвращается ICMPv6-сообщение об ошибке, опять-таки позволяющее отличить отсутствующие порты от портов, закрытых брандмауэром.

Протокол SMB, обеспечивающий совместный доступ к файлам и принтерам, так же полностью переписан и представлен в новой версии как SMB2, ориентированный на передачу больших файлов данных и как будто бы обеспечивающий лучшую производительность, однако реализованный далеко не самым лучшим образом. В частности, засылка мусора в порт 445 обрушивала бету build 5270 в голубой экран смерти, и этот косяк был исправлен только в следующей версии [2].

Механизмы аутентификации, вызывающие множество нареканий еще со времен 9x, похоже, не претерпели никаких радикальных изменений, откатившись назад в мрачную готическую тьму средневековья, когда нестандартные клиенты типа SAMBA предоставляли доступ ко многим защищенным ресурсам, не требуя авторизации. Помнится, реакция Microsoft была такова: «SAMBA – это неправильный клиент, пользуйтесь штатными средствами Windows, и у вас не будет никаких проблем». Протокол SMB держит для своих внутренних целей именованный канал (по-английски – pipe) «IPC$», через который можно подключаться к ресурсам netlogon, lsarpc и samr БЕЗ аутентификации! В SMB2 этот список пополнился каналами «protected_storage» и «lsass».

Механизм именованных каналов тесно связан со столь горячо любимым в Microsoft механизмом удаленного вызова процедур Remote Procedure Call или, сокращенно, RPC, через который распространялся MSBlast и другие черви подобного типа. В Висте до сих пор сохранилась возможность определять список доступных интерфейсов и вызывать некоторые из них (ServerAlive2, OXIDResolver, etc), и все это – без всякой авторизации!

## Глобализация ARP

В локальных Ethernet-сетях на физическом уровне используется MAC-адресация, поверх которой натягивается TCP/IP, использующий IP-адресацию. В результате этого каждый узел имеет как минимум один MAC-адрес и один IP-адрес, которые никак не связаны с друг другом, и чтобы отправленный пакет дошел до места назначения, маршрутизатору необходимо иметь таблицу соответствия IP- и MAC-адресов, которая динамически создается при помощи протокола ARP. Грубо говоря, в сеть посылается широковещательный запрос: «Обладатель такого-то IP, сообщите своей MAC-адрес!». Никакой аутентификации при этом не производится, и присвоить себе чужой IP – плевое дело. Атаки такого типа давно изучены и подобно описаны. Хакер может: разрывать TCP-/IP-соединения, установленные жертвой, перехватывать трафик, выдавать себя за другой узел и прочее. Но все это – строго в рамках локальной сети, причем предыдущие версии Windows, обнаружив, что хакер захватил их IP-адрес, выплевывали на экран предупреждение. Виста же просто отмечает этот факт в системном журнале и... прекращает реагировать на сетевые запросы.

Существует масса способов вычислить злоумышленника, подняв по тревоге бригаду каратистов быстрого реагирования, доходчиво объясняющих незадачливому хакеру, что лучше уйти по-хорошему, чем всю жизнь работать на больницу. Заботясь о пользователях, тьфу, о хакерах, Microsoft добавила новый протокол для разрешения ад-

ресов – Neighbor Discovery или сокращенно ND, доступный извне локальной сети. И хотя реализация удаленной атаки сопряжена с рядом трудностей, она все-таки осуществима! Система уязвима только во время так называемой probe-фазы, в течение которой происходит ожидание отклика от «соседних» (neighbor) узлов. Все остальное время поддельные пакеты, посланные злоумышленником, игнорируются. Однако, учитывая значительную продолжительность probe-фазы, а также ее высокую периодичность, хакеру даже не понадобится запасаться терпением! Ну, разве за пивом сгонять, пока его компьютер методично бомбардирует жертву запросами.

Трудность номер два: ND-пакет содержит специальный счетчик, начальное значение которого равно 255, и при пересылке через каждый узел оно уменьшается на единицу, таким образом, атакующий должен находиться достаточно близко от жертвы. «Близко», естественно, не в географическом смысле.

## Идентификация сетевого стека Висты

Сетевой стек всякой операционной системы имеет свои особенности реализации (они же «fingerprint» – отпечатки пальцев), позволяющие идентифицировать жертву, что значительно упрощает атаку, поскольку хакер заранее знает, какие дыры там есть и какие действия предпринимать. Снять отпечатки пальцев (также называемые «сигнатурой») с удаленного узла можно, например, с помощью знаменитой утилиты nmap.

## Заключение

ОС Windows Vista являет собой наиболее существенное обновление сетевых возможностей операционной системы за весь период времени, прошедший после выпуска Windows 95. Пользователям стало проще использовать преимущества проводных и беспроводных сетей в своих поездках. Благодаря новому самонастраиваемому сетевому стеку ускоряется передача файлов по сети.

Вопреки всем заверениям Microsoft, сетевой стек ОС Windows Vista намного менее надежен и безопасен, чем в XP, к тому же он совершенно не изучен и абсолютно непредсказуем. У администраторов нет опыта решения проблем, с которыми они прежде не сталкивались, разработчики защитных компонентов (от программных брандмауэров до аппаратных комплексов) еще не включили поддержку ОС Windows Vista и ее протоколов в свои продукты. Переход на ОС Windows Vista, несомненно, сулит большие перспективы для хакеров, а также для всех сторонних разработчиков, предлагающих защитные комплексы разной степени сложности.

## Литература

1. http://www.rfc-editor.org/rfc/rfc4380.txt
2. http://www.microsoft.com/technet/prodtechnol/winxppro/maintain/teredo.mspx
3. http://www.symantec.com/avcenter/reference/ATR-VistaAttackSurface.pdf

# ЗАКОНОДАТЕЛЬНЫЕ ТРЕБОВАНИЯ В ОБЛАСТИ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ АВТОМАТИЗИРОВАННЫХ СИСТЕМ

**А.Л. Липатов, Д.В. Осломенко**
**Научный руководитель – д.т.н., профессор Ю.А. Гатчин**
**М.В. Масленников**
**Научный руководитель – д.т.н., профессор А.Г.Коробейников**

В статье проводится анализ действующих законодательных требований Российской Федерации в области обеспечения информационной безопасности. Даются практические рекомендации по обеспечению конфиденциальности, целостности и доступности конфиденциальной информации, обрабатываемой в автоматизированных системах

## Введение

Сегодня стопроцентная зависимость бизнес-процессов организации от информационных технологий (ИТ) – не редкость. Однако повсеместное использование ИТ не только предоставило организациям значительные выгоды, но и сделало бизнес более уязвимым. В рамках общего процесса управления рисками (кредитными, финансовыми, операционными) организации вынуждены оценивать, в том числе, и риски информационной безопасности (ИБ) и предпринимать соответствующие действия, направленные на их обработку. С чего же необходимо начинать и как обеспечить организации требуемый уровень ИБ?

## Обеспечение информационной безопасности автоматизированных систем с учетом требований законодательства Российской Федерации

Существует классический поход к созданию систем обеспечения информационной безопасности (СОИБ) автоматизированных систем (АС), включающих в себя комплекс организационных мер и программно-аппаратных средств ИБ, описанный в действующих нормативно-методических документах. Выделяют четыре стадии создания и эксплуатации СОИБ АС.

Первая стадия – предпроектная. Начинать необходимо с уточнения состава, конфигурации и топологии АС, условий ее расположения, перечня используемых технических средств и программного обеспечения (ПО), субъектов и объектов доступа, анализа реализованных мер по обеспечению ИБ АС.

Важным этапом создания СОИБ АС (с точки зрения руководящих документов ФСТЭК (Гостехкомиссии) России) является классификация защищенности АС от несанкционированного доступа (НСД) к информации. Классификация проводится для всех АС, предназначенных для обработки конфиденциальной информации, в соответствии с руководящим документом «Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования по защите информации» (Гостехкомиссия России, 1992 г.) [1]. В результате классификации защищенности АС определяются базовые меры по обеспечению ИБ, окончательный состав которых формируется на этапе обработки рисков ИБ. Возможные классы защищенности АС приведены в таблице.

После уточнения ИТ-инфраструктуры и анализа реализованных мер по обеспечению ИБ АС должна быть произведена идентификация защищаемых активов и оценка их важности. К защищаемым активам относятся информационные активы (информация, обрабатываемая в АС), функциональные активы (сервисы АС), а также физические активы (критичные компоненты инфраструктуры АС – сетевое оборудование, серверы,

ПО и т.п.). При этом, если конфиденциальная информация, обрабатываемая в АС, содержится в государственных информационных ресурсах или относится к категории персональных данных, то установленные требования по ИБ будут носить обязательный характер. Если же информационные ресурсы являются негосударственными, то требования руководящих документов ФСТЭК (Гостехкомиссии) России и ФСБ России по защите конфиденциальной информации будут являться не более чем рекомендациями.

| | Класс защищенности однопользовательской АС | Класс защищенности многопользовательской АС с равными правами пользователей по доступу к активам | Класс защищенности многопользовательской АС с различными правами пользователей по доступу к активам |
|---|---|---|---|
| Служебная тайна | не ниже 3Б | не ниже 2Б | не ниже 1Г |
| Коммерческая тайна, банковская тайна и т.п. | не ниже 3Б | не ниже 2Б | не ниже 1Д |
| Персональные данные | не ниже 3Б | не ниже 2Б | не ниже 1Д |

Таблица. Классы защищенности АС от НСД к информации

Далее производится идентификация рисков ИБ, а также их оценка (количественная или качественная). Идентификация рисков ИБ производится на основе анализа актуальных угроз ИБ и выявленных уязвимостей как технического, так и организационного характера. Оценка рисков осуществляется на основе анализа уровней угроз, уязвимостей с учетом ценностей активов.

В завершение первой стадии проводится обработка идентифицированных рисков ИБ в части формирования предложений по обеспечению требуемого уровня ИБ АС. Обработка рисков заключается в выборе действий по модификации рисков. Существуют следующие возможные действия: снижение рисков путем применения подходящих защитных мер, принятие рисков, избежание рисков и перенос рисков на другие стороны, например, на страховщиков или поставщиков.

Предпроектное обследование АС может выполняться как непосредственно специалистами отдела ИБ организации, так и специализированной организацией, имеющей Лицензию ФСТЭК (Гостехкомиссии) России на деятельность по технической защите конфиденциальной информации. Во втором случае затраты на оплату услуг специализированной организации должны окупиться за счет независимости, полноты и качества проведенных работ. Конечное же решение по выбору исполнителя остается за руководством организации.

После проведения предпроектного обследования формируется техническое задание на СОИБ АС.

Работы по разработке документации и проектированию СОИБ АС, установке и настройке в соответствии с политиками безопасности соответствующих программно-технических средств ИБ выполняются на второй стадии создания и эксплуатации СОИБ АС. Технические, программные и организационные решения по обеспечению ИБ разрабатываются в Техническом проекте и рабочей, организационно-распорядительной и эксплуатационной документации на СОИБ АС.

К основным мерам по обеспечению ИБ АС относятся:

- кадровая работа, исключающая прием на работу «ненадежных» людей;
- реализация допуска исполнителей к конфиденциальной информации;

- организация строгого контрольно-пропускного режима в здание организации и помещения АС. Применение технических средств охраны, контроля и управления доступом;
- обучение пользователей, администраторов и обслуживающего персонала;
- разделение ролей и обязанностей в области ИТ и ИБ;
- постоянный анализ внутренних и внешних угроз ИБ, мониторинг событий ИБ, расследование инцидентов ИБ;
- надлежащий учет, маркировка, обращение и хранение носителей конфиденциальных информации в соответствии с установленными требованиями;
- расположение экранов мониторов, исключающее несанкционированный просмотр информации с них;
- организация защиты информации от утечки по каналам побочных электромагнитных излучений и наводок, в т.ч. с использованием сертифицированных на соответствие требованиям по безопасности информации технических средств защиты (например, ЛГШ-501, Гном-3, ФСП-1Ф-7А, ЛФС-10-1Ф, ЛГШ-220 и т.п.);
- резервное копирование критичной информации и ПО;
- использование средств защиты от вредоносного ПО. Регулярное обновление баз средств защиты от вредоносного ПО в автоматическом режиме;
- оперативная установка на компьютеры всех последних программных обновлений (Service Pack, Patch);
- предоставление пользователям только необходимых полномочий и привилегий в системе;
- использование для аутентификации подходящих методов идентификации и аутентификации;
- организация разграничения доступа пользователей к конфиденциальной информации;
- использование для ограничения доступа со стороны сети Интернет к внутренним ресурсам сети, а также для разграничения доступа пользователей АС к внешним ресурсам средств защиты периметра сети - межсетевых экранов, систем ID&PS (систем обнаружения и предотвращения вторжений);
- использование для обеспечения целостности и безопасности, передаваемых по сети Интернет между удаленными филиалами (мобильными компьютерами) данных, технологии виртуальных локальных сетей (VPN). Применение в VPN-сетях стойких криптографических алгоритмов (например, ГОСТ 28147-89, AES и т.п.);
- использование сканеров уязвимостей для оперативного обнаружения и анализа сетевых уязвимостей сети в изменяющемся сетевом окружении (например, Internet Scanner, Nessus и т.п.).

После разработки проекта создания СОИБ АС, установки и настройки соответствующих средств ИБ, реализации предписанных организационных мер и разработки всех необходимых документов начинается стадия ввода в действие СОИБ АС. На этой стадии производится опытная эксплуатация установленных средств ИБ в комплексе с общесистемным и прикладным ПО и техническими средствами АС, а также аттестация АС на соответствие требованиям по безопасности конфиденциальной. Для АС, обрабатывающих конфиденциальную информацию, содержащуюся в государственных информационных ресурсах, аттестация носит обязательный характер и должна предшествовать началу обработки защищаемой информации. Для негосударственных организаций аттестация носит добровольный характер.

И, наконец, четвертая стадия – стадия эксплуатации СОИБ АС. На этой стадии требуется проводить регулярный анализ эффективности СОИБ АС, анализировать уровень остаточных рисков ИБ, проводить внутренний аудит, регистрировать действия и события, влияющие на СОИБ АС, реализовать улучшения, предпринимать корректи-

рующие/предупреждающие действия, а также информировать о результатах все заинтересованные стороны.

## Заключение

Работы по обеспечению ИБ АС должны носить комплексный и системный характер, выполняться профессионально и качественно. Только в этом случае удастся обеспечить информационную безопасность не только де-юре, но и де-факто. При этом реализуемые в рамках СОИБ АС защитные меры должны быть экономически обоснованы и выбираться на основе результатов оценки рисков ИБ.

## Литература

1. Руководящий документ «Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования по защите информации» (Гостехкомиссия России, 1992 г.). С. 2–3.

# ОСНОВНЫЕ АСПЕКТЫ СОЗДАНИЯ СИСТЕМЫ ЗАЩИТЫ ПЕРИМЕТРА КОРПОРАТИВНОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ

**Н.В.Ермаков, К.В.Строганов**
**Научные руководители – д.т.н., профессор Ю.А. Гатчин, д.т.н., профессор А.Г. Коробейников**

Цель статьи – осветить основные аспекты, необходимые для создания системы безопасного взаимодействия корпоративной информационной системы предприятия с единой вычислительной сетью передачи данных и с глобальной сетью Интернет.

## Введение

В процессе взаимодействия корпоративной информационной системы (далее КИС) предприятия с единой вычислительной сетью передачи данных (далее ЕВСПД) объективно существует возможность:

1. несанкционированного доступа внешних и внутренних пользователей к информационным ресурсам КИС и внутренних пользователей к внешней сети;
2. получения информации о структуре локальной вычислительной сети и ее компонентов внешними пользователями;
3. атак, нарушающих или создающих предпосылки к нарушению требований по безопасности информации КИС.

Исходя из этого, должен быть, создан комплекс средств защиты периметра корпоративной информационной системы (далее – КСЗП КИС) предприятия. Разработка КСЗП КИС должна проводиться на основании документов, регламентирующих информационную безопасность всей распределенной сети передачи данных в целом и корпоративной информационной системы предприятия, в частности.

## Цели обеспечения безопасности информации в КИС предприятия

Основной целью обеспечения защиты информации в КИС являются нейтрализация потенциальных угроз информационной безопасности и предотвращение нанесения ущерба в экономической сфере, внутриполитической, международных сферах, а также в области науки и техники в результате возможной утечки информации, несанкционированного или непреднамеренного воздействия на нее и/или на информационные системы. Для формирования грамотного технического решения, необходимо провести комплексный анализ всех факторов влияющих на информационную безопасность предприятия. Основной целью работ по анализу угроз является идентификация существующих уязвимостей информационной системы, идентификация угроз нарушения информационной безопасности и оценка результатов их потенциального воздействия, как на обследуемую информационную систему организации, так и на деятельность организации в целом. Анализ проводится для оценки угроз нарушения информационной безопасности и разработки рекомендаций по устранению выявленных уязвимостей в КИС предприятия.

Анализ угроз дает возможность:

- идентифицировать критичные ресурсы информационной системы;
- адекватно оценить существующие угрозы;
- сформировать перечень наиболее опасных уязвимых мест, угроз и потенциальных злоумышленников;
- выработать адекватные требования по защите информации;

- получить определенный уровень гарантий, основанный на объективном экспертном заключении.

В ходе анализа осуществляется:

1. классификация информационных ресурсов;
2. составление модели источника угроз (в том числе модели потенциального нарушителя);
3. идентификация и оценка угроз нарушения информационной безопасности;
4. анализ уязвимостей;
5. разработка рекомендаций по устранению выявленных уязвимостей.

### Классификация информационных ресурсов

Проведенный анализ КИС предприятия позволит выделить объекты (физические, информационные) и субъекты защиты информации и их характеристики. Результаты анализа могут быть представлены в виде табл. 1–3.

| Структурный компонент | Тип объекта защиты | Наименование объектов защиты | Функции объекта защиты | Расположение |
|---|---|---|---|---|
| КИС предприятия | 1.АРМ | 1.1.АРМ КИС | | |
| | 2.Серверы | 2.1. Серверное оборудование | | |
| | 3.Коммуникационное оборудование | Концентраторы/коммутаторы | | |
| | | 3.1. Маршрутизаторы | | |
| | | 3.2. Модемы | | |
| | | 3.3. Кабельные каналы и трассы | | |
| | | 3.4. Сетевые розетки | | |

Таблица 1. Физические объекты защиты

| Класс информационного объекта защиты | Наименование информационного объекта защиты | Принадлежность физическому объекту защиты | Носитель информации |
|---|---|---|---|
| Данные, выдаваемые в канал связи | 1.1.Документы (в том числе электронные письма) | | |
| | 1.2.Запросы БД | | |
| | 1.3.Отчеты БД | | |

Таблица 2. Информационные объекты защиты

| Уровень возможностей нарушителя | Субъекты информационной безопасности | Режим работы | Объект защиты |
|---|---|---|---|
| Первый уровень – возможность запуска программ из фиксированного набора, реализующих заранее предусмотренные функции по обработке информации | Пользователь АРМ, пользователь сети | | |
| Второй уровень – возможность создания и запуска собственных программ с новыми функциями по обработке информации | Прикладной программист разработчик программного обеспечения | | |

Таблица 3. Возможная модель нарушителя

| Уровень возможностей нарушителя | Субъекты информационной безопасности | Режим работы | Объект защиты |
|---|---|---|---|
| Третий уровень – возможность получения управления функционированием системы, а также воздействия на базовое программное обеспечение, состав и конфигурацию оборудования | Системный программист, администратор сервера (ЛВС), администратор информационной системы (базы данных), разработчик | | |
| Четвертый уровень – возможность проектирования, установки и ремонта средств электронно-вычислительной техники, вплоть до включения в их состав собственных технических и программных средств с новыми функциями по обработке информации | Администратор информационной системы, администратор сервера (ЛВС), администратор безопасности информации, разработчик системы, разработчик средств защиты информации, обслуживающий персонал | | |

Таблица 3 (продолжение). Возможная модель нарушителя

**Решения по структуре КСЗП КИС предприятия**

Для защиты информации КИС предприятия можно руководствоваться стратегией защиты информации на основе анализа модели нарушителя.

Основными классами угроз информационным ресурсам КИС предприятия являются нарушение целостности, доступности и конфиденциальности информации, обрабатываемой и хранящейся на серверах различных сегментов ЛВС, входящих в КИС предприятия. На втором месте находятся угрозы АРМам КИС предприятия, на которых обрабатывается конфиденциальная и критическая информация. Для серверов и АРМ КИС предприятия эти угрозы исходят, в основном, от внутренних пользователей сегментов КИС, а также со стороны сети ЕВСПД, необходимой для информационного взаимодействия между географически распределенными филиалами.

Для защиты от указанных выше основных и прочих угроз необходимо:

1. для управления КСЗП КИС предприятия использовать АРМ администратора безопасности, который предполагается расположить в ЛВС предприятия;
2. организовать физическое разделение ЛВС, выделив сегмент демилитаризованной зоны;
3. в КИС предприятия организовать защиту информации, для чего:

- обеспечивать фильтрацию входящего/исходящего трафика в соответствии с разработанными политиками доступа;
- выполнять трансляцию сетевых адресов для сокрытия внутренней структуры КИС;
- уведомлять администратора безопасности о попытках несанкционированных действий и других подозрительных событиях;
- регистрировать события с задаваемым уровнем детализации;
- выявлять атаки на информационные ресурсы КИС;
- уведомлять об атаке путем выдачи сообщения администратору безопасности;
- регистрировать события о выявленных атаках в базе данных;
- предотвращать атаки путем блокировки, завершения сессии с атакующим узлом и управления настройками межсетевого экрана;
- обеспечивать авторизованный защищенный доступ к средствам межсетевого экранирования, обнаружения и предотвращения вторжений.

Для выполнения вышеперечисленного необходимо на границе между КИС и ЕВСПД установить межсетевые экраны с глубокой инспекцией пакетов для разграничения доступа в/из КИС предприятия и контроля информационного обмена путем дополнительной аутентификации пользователей

Для реализации структуры КСЗП КИС предприятия в виде конкретных технических решений можно произвести ее декомпозицию на подсистемы. Выделим в КСЗП КИС предприятия следующие подсистемы:

1. подсистема межсетевого экранирования;
2. подсистема обнаружения и предотвращения вторжений;
3. подсистема управления защитой периметра КИС;

Опишем в отдельности каждую подсистему и уточним выполняемые ими функции.

**Подсистема межсетевого экранирования**

Подсистема межсетевого экранирования предназначена для защиты информации объектов КИС предприятия от несанкционированного удаленного доступа, сокрытия информации о структуре ЛВС и ее компонентов от внешних пользователей, разграничения доступа из КИС во внешнюю сеть и из внешней сети в КИС.

Подсистема должна выполнять следующие функции:

- фильтрация входящего/исходящего трафика в соответствии с разработанными политиками доступа;
- идентификация и аутентификация пользователей на основе имени пользователя и его учетной записи;
- трансляция сетевых адресов для сокрытия внутренней структуры КИС;
- уведомление администратора безопасности о попытках несанкционированных действий и других подозрительных событиях;
- регистрация событий с задаваемым уровнем детализации;
- организация демилитаризованной зоны;
- резервирование с возможностью восстановления.

**Подсистема обнаружения и предотвращения вторжений**

Подсистема обнаружения и предотвращения вторжений предназначена для обнаружения и предотвращения атак, нарушающих или создающих предпосылки к нарушению требований по безопасности информации в КИС. Средствами подсистемы обнаружения и предотвращения вторжений должны быть реализованы следующие функции:

- выявление атак на информационные ресурсы КИС;
- уведомление об атаке путем выдачи сообщения администратору безопасности;
- регистрация событий об выявленных атаках в базе данных;
- предотвращение атак путем блокировки работы атакующего, завершения сессии с атакующим узлом и управления настройками межсетевого экрана.

**Подсистема управления защитой периметра КИС**

Подсистема управления защитой периметра КИС предприятия предназначена для управления программными и техническими компонентами подсистем КСЗП, установленными в рамках данного проекта. Средства подсистемы должны обеспечивать возможность управления средствами, входящими в состав КСЗП. Подсистема управления защитой периметра должна реализовывать следующие функции:

- авторизованный защищенный доступ к средствам межсетевого экранирования, обнаружения и предотвращения вторжений;

- настройка политик безопасности для подсистем межсетевого экранирования, обнаружения и предотвращения вторжений;
- сбор статистики;
- генерация отчетов;
- управление средствами межсетевого экранирования и обнаружения и предотвращения вторжений.

## Заключение

В результате проведенной декомпозиции по подсистемам выделяется перечень функций, которые должны обеспечиваться техническими решениями, применяемыми в КСЗП КИС предприятия. Выбор технических решений по выявленным подсистемам остается за рамками данной статьи, так как требует углубленного рассмотрения соответствия предлагаемых на рынке программно-аппаратных средств защиты требованиям к конкретной подсистеме.

## Литература


1. Концепция защиты СВТ и АС от НСД к информации. 1998 г.
2. Защита от несанкционированного доступа к информации. Термины и определения. 1998.
3. Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования по защите информации. 1998.
4. Средства вычислительной техники. Защита от несанкционированного доступа к информации. Показатели защищенности от НСД к информации. 1998.
5. Средства вычислительной техники. Межсетевые экраны. Защита от несанкционированного доступа к информации. Показатели защищенности от несанкционированного доступа к информации. 1998.

# ПРИМЕНЕНИЕ НЕЙРОННОЙ СЕТИ ДЛЯ ОБНАРУЖЕНИЯ СЕТЕВЫХ АТАК

**Э.В. Белов, М.В. Масленников**
**Научный руководитель – д.т.н., профессор А.Г. Коробейников**

В статье рассматривается возможность применения нейронных сетей для обнаружения сетевых атак. Благодаря таким свойствам, как адаптивность, способность анализа по неполным или искаженным данным, высокая скорость обработки данных, нейронные сети могут существенно улучшить процесс обнаружения и предотвращения атак.

## Введение

Большинство современных подходов к обнаружению атак используют различные формы анализа на основе правил. Анализ на основе правил опирается на набор заранее предопределенных условий, которые вводятся экспертом, автоматически создаются системой, или используются оба варианта. Экспертные системы (ЭС) представляют собой наиболее распространенный подход к обнаружению атак на основе правил. Первоначальные усилия по исследованию обнаружения атак показали неэффективность любого подхода, который требует ручного просмотра журнала регистрации событий.

Информация, необходимая для идентификации атак, представлена большим количеством данных аудита, и эффективный обзор этих данных требовал использования автоматической системы для их анализа. Применение ЭС в системах обнаружения атак способствовало разработке эффективных систем информационной безопасности. ЭС состоит из набора правил, которые охватывают знания человека-эксперта. Эти правила используются системой для выявления злонамеренной деятельности в данных, получаемых от системы обнаружения атак. ЭС допускают объединение огромного опыта, накопленного человеком, в компьютерном приложении, которое затем использует эти знания для выявления деятельности атак. К сожалению, ЭС требуют постоянного обновления для поддержания актуальности.

Основные недостатки применения экспертных методов в СОА:

- низкая возможность выявления атак, которые имеют место в продолжительных периодах времени или различных согласовано действующих источниках;
- неспособность выявления малоизвестных или отличающихся от шаблона правил СОА атак.

В последние годы разработано большое количество подходов к обнаружению атак, отличных от ЭС. Несмотря на то, что многие из них выглядят довольно многообещающими, ЭС остаются наиболее распространенным подходом к обнаружению атак.

## Нейронные сети

Искусственная нейронная сеть (artificial neural network) (НС) состоит из набора элементарных элементов, которые взаимосвязаны друг с другом и преобразуют набор входных данных к набору желаемых выходных данных. Результат преобразования определяется характеристиками элементов и весами, соответствующими взаимосвязям между ними [1]. Путем видоизменения соединений между узлами сети можно адаптироваться к желательным выходным результатам.

В отличие от ЭС, определяющих соответствие конкретным характеристикам, заложенным в базе данных правил, НС проводит анализ информации и предоставляет возможность оценить, что данные согласуются с характеристиками, которые она научена распознавать. В то время как степень соответствия нейросетевого представления может достигать 100 %, достоверность выбора полностью зависит от качества системы

в анализе примеров поставленной задачи (так называемое обучение). Первоначально НС обучается путем правильного выявления предварительно отобранных данных предметной области. Реакция НС анализируется и настраивается в системе таким образом, чтобы достичь удовлетворительных результатов. В дополнение к первоначальному периоду обучения, НС также набирается опыта с течением времени по мере того, как она проводит анализ данных, связанных с предметной областью.

По вопросу применения НС для обнаружения компьютерных атак было проведено большое количество исследований. Искусственные НС предлагают потенциал для решения большого количества проблем, охватываемых другими современными подходами к обнаружению атак. Искусственные НС были предложены в качестве альтернативы компонентам статистического анализа систем обнаружения аномалий. Статистический анализ включает статистическое сравнение текущих событий с предварительно определенным набором эталонных критериев. Этот метод наиболее часто используется при обнаружении отклонений от типичного режима и определяет события, аналогичные событиям, которые указывают на атаку. НС были специально предложены для того, чтобы классифицировать типичные действия пользователей системы и выявить статистически значимые отклонения от установленного режима работы пользователя.

Системы обнаружения злоупотреблений на основе НС способны решить некоторые проблемы, имеющиеся в системах на основе правил. Основные преимущества применения НС:

- гибкость – возможность выявления злоупотреблений по неполным или искаженным данным;
- способность выявлять малоизвестные атаки, а также вероятность проведения атаки распределенной во времени;
- высокая скорость анализа данных.

К недостаткам следует отнести:

- необходимость обучения НС, что обусловливает применение различных методов обучения и подготовки обучающих данных (выборки) для наилучшего анализа и выявления злонамеренной деятельности;
- суть происходящих внутри НС процессов скрыта, и качество анализа зависит непосредственно от обучения.

Есть несколько распространенных вариантов реализации НС в СОА. Первый включает объединение их с существующими экспертными системами (рис. 1). Данное решение использует НС для фильтрации входящих данных, которые могут указывать на злоупотребления, и передачи этих событий к экспертной системе. Эта конфигурация должна улучшить эффективность системы обнаружения за счет снижения числа ложных срабатываний, присущих ЭС. Поскольку НС будет определять вероятность того, что конкретное событие указывает на атаку, пороговая величина может быть установлена там, где событие перенаправляется к ЭС для дополнительного анализа.

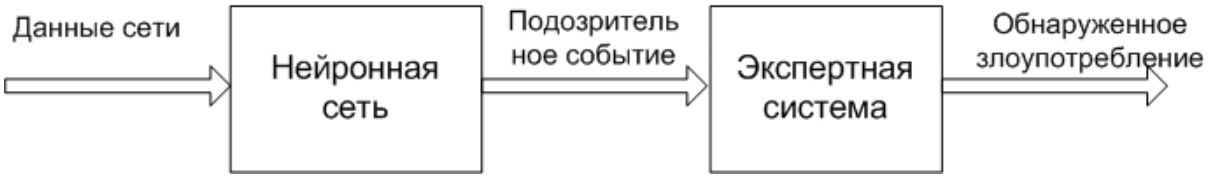



Рис. 1. Схема совместного использования НС и экспертной системы

Основной недостаток данного подхода заключается в необходимости поддержания актуальности баз данных ЭС в соответствии с уровнем обучения НС. Если экспертная система не была обновлена, то новые атаки, выявляемые НС, будут в значительной степени пропускаться ЭС, потому что ее собственные правила не способны распознать новую угрозу.

Второй подход заключается в реализации НС как отдельной системы обнаружения злоупотреблений (рис. 2). В этой конфигурации НС получает весь поток данных и анализирует информацию на наличие в них злоупотреблений. Любые случаи, которые идентифицируются с указанием на атаку, перенаправляются к администратору безопасности или используются системой автоматического реагирования на атаки. Основные преимущества данного подхода:

- высокая скорость выявления атак по сравнению с предыдущим подходом, так как существует только один уровень анализа;
- повышение эффективности выявления атак с течением времени ввиду обучения НС. В отличие от первого подхода, эта концепция не ограничивается аналитической способностью ЭС.

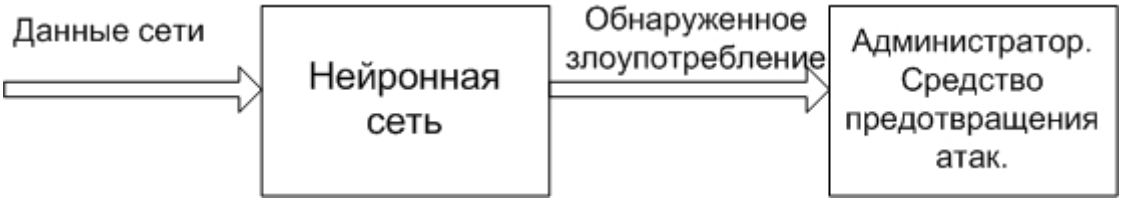



Рис. 2. Схема применения НС для обнаружения злоупотреблений

**Практическое применение НС**

НС получили практическое применение во многих исследованиях систем обнаружения атак. Так, в работах Дж. Райана (J.Ryan) [2] описывается автономная система выявления аномалий (Off-line anomaly detection system), в которой применяется многослойная НС, обучаемая по алгоритму обратного распространения ошибки. Данная сеть обучалась профилю пользователя, работающего на компьютере, выявляя в командах пользователя возможные отклонения (аномалии). Для НС была выбрана 3-слойная структура с двумя скрытыми слоями. НС позволила выявить аномалии в 22 случаях из 24.

В работе Джеймса Кеннеди (James Cannady) [3] 3-слойная НС применялась для автономной классификации записей сетевых соединений по классам нормальной и подозрительной деятельности. В работе использовалась выборка из 10000 записей сетевых соединений, из которых 1000 записей являлись имитированными сетевыми атаками. В процессе обучения использовалась выборка из 30 % записей. В итоге полученная система позволила правильно классифицировать подозрительную деятельность в 89–91 % случаях. В других исследованиях [4] авторы применили 3-слойные и 4-слойные НС, получив результаты определения подозрительной деятельности в 99,25 % случаях. Различные группы исследователей использовали в своей работе самоорганизующиеся карты(Self-Organized Maps) для обнаружения атак [5].

**Заключение**

В связи с особенностями применения НС реализация систем обнаружения злоупотреблений реального времени, основанная исключительно на данном подходе, в практическом плане весьма затруднительна, что не исключает применения НС в автономных(off-line) системах. Необходимость обучения, а также природа «черного ящика» НС обуславливает обязательное наличие обучающей базы данных злоупотреблений, а также временных затрат и корректировок обучающего процесса (выбора архитектуры сети, алгоритма обучения и т.д.) В дальнейших исследованиях следует уделить внимание совместного применения различных техник искусственного интеллекта (нечеткая логика, нейронные сети и т.д.).

## Литература


1. Заенцев И.В. Нейронные сети: основные модели. Изд. Воронежского ГУ, 2001.
2. J. Ryan, M. Lin, and R. Miikkulainen. Intrusion Detection with Neural Networks. / AI Approaches to Fraud Detection and Risk Management: Papers from the 1997 AAAI Workshop, Providence, RI. P. 72–79, 1997.
3. James Cannady, Artificial neural networks for misuse detection. / Proceedings of the 1998 National Information Systems Security Conference (NISSC'98), Arlington, VA, 1998.
4. Srinivas Mukkamala, Intrusion detection using neural networks and support vector machine. / Proceedings of the 2002 IEEE International Honolulu, HI, 2002.
5. P. Lichodzijewski, A.N. Zincir Heywood, and M. I. Heywood, Host-based intrusion detection using self-organizing maps. / Proceedings of the 2002 IEEE World Congress on Computational Intelligence, Honolulu, HI, pp. 1714–1719, 2002.

# МЕТОДИКИ РАЗРАБОТКИ ЗАЩИЩЕННОЙ СИСТЕМЫ АВТОМАТИЗАЦИИ УПРАВЛЕНИЯ ПРОМЫШЛЕННЫМ ПРЕДПРИЯТИЕМ

**М.В. Барышев, А.А. Гуськов**
**Научный руководитель – д.т.н. профессор Ю.А. Гатчин**

При создании автоматизированной системы управления предприятием, базирующейся на использовании технологии web-приложения, необходимо учитывать множество особенностей, связанных с вопросами информационной безопасности. Задача становится более сложной, когда идет речь об использовании в качестве платформы мобильной системы. Разнообразие аппаратных и программных средств должно поддерживаться для осуществления максимальной доступности и отказоустойчивости системы при различных сдерживающих факторах, не связанных с информационной безопасностью. В работе предложен метод создания защищенного приложения на базе мобильного устройства.

## Введение

Уже несколько десятков лет в промышленности внедряются и активно используются системы автоматизации производственных процессов. В информационный век все чаще приходится задумываться не только о том, как автоматизировать процессы, но и как получать оперативную и достоверную информацию об этих процессах. Эту задачу способны решать современные технологии (такие как глобальные сети и беспроводные средства связи). Но важно не забывать о безопасности получения подобного рода сведений.

С проблемами промышленного шпионажа приходится сталкиваться в современном мире все чаще. Внедряются более сложные устройства и более современные технологии, чтобы встать на защиту коммерческой тайны и обеспечить экономическую безопасность. Но вместе с тем и нарушитель становится более технологически подготовленным и, в свою очередь, делает все, чтобы получить необходимые сведения. Для этого злоумышленник использует различные методы, начиная от психологических уловок и социальных технологий и заканчивая применением последних достижений техники.

Абсолютной и идеальной защиты не существует, поэтому мы можем лишь разрабатывать все более сложные или «умные» средства и способы защиты. Необходимо также помнить, что каким бы технически совершенным ни было наше оборудование и система защиты, все это оказывается бессмысленным без учета человеческого фактора. Именно человеческий фактор был и остается одним из самых уязвимых мест. Поэтому важно постоянно проводить организационные мероприятия, повышающие уровень безопасности системы, постоянно инструктировать персонал по грамотному использованию системы безопасности.

## Основная часть

В данной статье будут подробно рассматриваться механизм создания защищенного приложения автоматизированной системы управления предприятием для мобильных платформ. Схема автоматизированной системы управления предприятием базируется на Web-приложении типа «клиент-сервер», использует защищенную базу данных и доступна для клиентов, имеющих доступ к глобальной сети Internet (рис. 1).

Серверная часть системы состоит из:

- сервера приложений, который содержит Web-сервер, а также модули, обеспечивающие защиту канала передачи данных посредством организации виртуальной частной сети (VPN), а также обеспечивающий шифрование трафика;
- сервера базы данных, которая обеспечивает ответы на запросы пользователя посредством сервера приложений, а также безопасное хранение данных.

Клиентская часть системы может быть запущена на стационарной платформе (персональный или промышленный компьютер, ноутбук и т.п.), а также организована на базе мобильной платформы (карманный компьютер, коммуникатор или мобильный телефон). В любом варианте платформа клиента должна поддерживать запуск следующих приложений:

- Web-приложение для осуществления запросов к базе данных;
- приложение, обеспечивающее защиту канала передачи данных посредством организации виртуальной частной сети (VPN);
- приложение, обеспечивающее шифрование трафика.

Устройство должно иметь возможность подключения к глобальной сети Internet, содержать устройство чтения карт памяти типа Secure Digital, а также модуль Bluetooth.

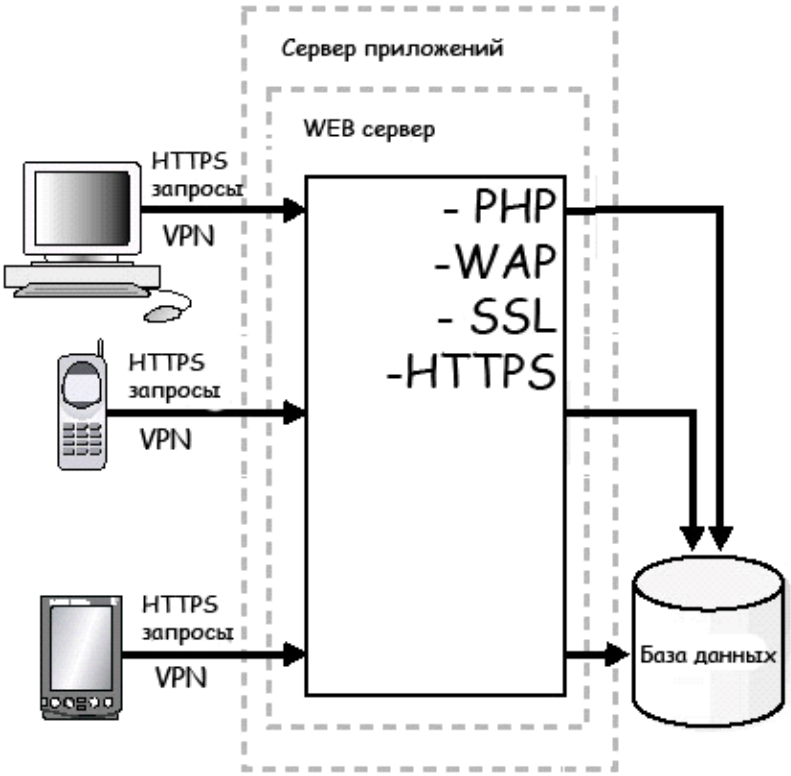



Рис. 1. Общая схема автоматизированной системы управления предприятием

Безопасность работы с платформой приложения обеспечивается следующими организационно-техническими мерами:

- встроенная защита устройства (для персональных компьютеров – средства авторизации операционной системы, жесткого диска, BIOS; для мобильных телефонов – PIN-код доступа SIM-карты, самого аппарата);
- организация виртуальной частной сети (VPN) в канале передачи данных (может быть реализована частично при помощи программных средств, частично при помощи сервисов компании, предоставляющей услуги связи);
- шифрование трафика;
- использование электронного ключа (хранимого на карте памяти устройства);
- использование защищенной базы данных (за счет встроенных средств сервера, обеспечивающих безопасность данных)
- использование Bluetooth-иммобилайзера;

- составление грамотной документации, которая будет четко регламентировать работу с защищенной системой, настройку и обслуживание критических модулей, а также поведение персонала в экстренных случаях;
- организационные меры обеспечения безопасности (обучение персонала, аудит действий пользователей в системе и т.п.).

Теперь рассмотрим подробнее схему защиты автоматизированной системы управления предприятием (рис. 2).

**Встроенная защита устройства.** Данный вид защиты имеется в составе любой платформы приложений, но не стоит особо рассчитывать на ее эффективность, так как она является лишь дополнением к системе безопасности и зачастую может быть вскрыта профессионалом в считанные минуты.

**Организация виртуальной частной сети (VPN) в канале передачи данных.** Для эмуляции частной связи данные шифруются для обеспечения конфиденциальности. Пакеты, проходящие публичные сети, являются нечитаемыми без ключей шифрования. Данная схема обеспечивает связь только с разрешенными узлами в сети. Она работает следующим образом.

- Удаленный клиент или шлюз инициирует соединение с сервером.
- Сервер посылает вызов клиенту.
- Клиент посылает шифрованный ответ серверу, который содержит имя пользователя и пароль.
- Сервер проверяет данные согласно своей базе данных.
- При корректных данных и успешной аутентификации сервер использует настройки соединения пользователя, авторизует соединение.
- Удостоверяющие данные используются только для обеспечения туннеля для удаленного доступа в сеть назначения. Клиент не заносится в сеть в результате установления соединения удаленного доступа. Каждый раз при попытке доступа в сетевые ресурсы он должен будет посылать удостоверяющие данные. Если они не соответствуют допустимым данным, попытка не будет завершена успешно.

**Шифрование трафика.** С развитием электронной коммерции появилась необходимость шифровать трафик, содержащий конфиденциальную информацию. Для шифрования трафика используется технология SSL (Security Socket Layer), работающая в реальном масштабе времени. Наибольшее развитие в настоящее время получило шифрование трафика программными средствами как наиболее дешевое и доступное. В данной схеме применяются протоколы шифрования на базе Web, которые обеспечивают доступ в Internet через TCP-порт 443, используемый шифрованными Web-страницами. Через этот же порт пересылается шифрованный Web-трафик, и многие администраторы брандмауэров оставляют этот порт открытым. Поэтому для организации данного протокола не требуется принимать специальных мер.

**Использование электронного ключа.** Уже много лет в качестве электронных ключей для персональных компьютеров используют usb-flash устройства. На это устройство записывается код, который необходим для дешифрования информации полученной с сервера. Этот код генерируется случайным образом и имеет большую длину, что исключает возможность его подбора. Эти устройства позволяют существенно повысить надежность аутентификации в самых различных приложениях. В данном проекте предлагается слегка изменить стандартную схему и использовать в качестве электронного ключа карту памяти Secure Digital. Такой вариант является более универсальным, так как его можно использовать не только в любых персональных компьютерах, но и в карманных компьютерах, коммуникаторах и мобильных телефонах. Электронные ключи выполнены в виде карты памяти, обеспечивают ряд преимуществ:

- для получения доступа к защищенной информации необходимо не только ввести пароль, но и подключить электронный ключ. Можно быть уверенным, что информация надежно защищена, если ключ находится у вас;
- один электронный ключ может использоваться сразу в нескольких приложениях и на различных платформах;
- код, хранимый на электронном ключе, генерируется случайным образом, имеет большую длину, что делает его подбор нереальной задачей;
- использование электронных ключей не требует запоминания сложного пароля;
- возможность доступа к защищенной информации может быть мгновенно заблокирована при извлечении электронного ключа.

**Использование защищенной базы данных.** Новая функциональная возможность сервера Oracle Database 10g Release 2 позволяет сделать следующее: когда пользователи вставляют данные, сервер базы данных прозрачно шифрует эти данные и сохраняет их в столбце. Точно так же, когда пользователи выбирают этот столбец, сервер базы данных автоматически расшифровывает его. Так как все это делается прозрачно без какого-либо изменения кода приложения, эта функциональная возможность имеет соответствующее название: прозрачное шифрование данных (TDE, Transparent Data Encryption). В среде сервера Oracle Database 10g Release 2 и шифрования TDE не нужно создавать эту инфраструктуру. Все, что требуется сделать – определить столбец, который будет шифроваться, и сервер создаст криптографически стойкий ключ шифрования для таблицы, содержащей этот столбец, и зашифрует данные обычного текста в этом столбце, используя указанный алгоритм шифрования. Защита этого ключа таблицы имеет очень важное значение; сервер шифрует его, используя главный ключ, хранящийся в безопасном месте, называемом бумажником (wallet), который может быть файлом сервера базы данных. Зашифрованные ключи таблиц размещаются в словаре данных. Когда пользователь вставляет данные в столбец, определенный как зашифрованный, сервер извлекает из бумажника главный ключ, расшифровывает ключ шифрования для этой таблицы, находящийся в словаре данных, использует этот ключ для шифрования входного значения и сохраняет зашифрованные данные в базе данных. Данные хранятся в зашифрованном виде, поэтому все компоненты нижнего уровня, такие, как резервные копии и архивные журнальные файлы, также имеют шифрованный формат. Когда пользователь выбирает зашифрованные столбцы, сервер прозрачно извлекает из словаря данных зашифрованный ключ таблицы, а главный ключ – из бумажника и расшифровывает ключ таблицы. Затем сервер базы данных расшифровывает зашифрованные на диске данные и возвращает пользователю обычный текст. Благодаря этой технологии, даже если данные будут украдены с диска, они не могут быть извлечены без главного ключа, который находится в бумажнике, не входящем в украденные данные. Если украден и бумажник, главный ключ не может быть извлечен из него без знания пароля бумажника. Следовательно, вор не сможет расшифровать данные, даже если он украл диски или копии файлов данных. Это удовлетворяет требованиям соответствия многим нормативным и руководящим документам. Все это было сделано без изменения приложения или написания сложной системы шифрования и управления ключами.

**Использование Bluetooth-иммобилайзера.** Подобные устройства очень активно используются как средства, обеспечивающие безопасность машин, домов и т.п., в данной системе реализована возможность использования подобной системы для обеспечения безопасности предлагаемой автором платформы приложения. Это средство, которое поможет обеспечить защиту данных даже в случае непредвиденных обстоятельств, в том числе нападения на пользователя системы, угрозы, физического овладения устройством с приложением в момент его использования, кражи и т.п. Устройство представляет собой мини-гарнитуру Bluetooth, которая должна находиться у пользователя и

быть в непосредственной близости от устройства платформы-приложения в момент осуществления связи с сервером. Она обеспечивает непрерывный контроль, чтобы в случае хищения устройства платформы приложения, когда злоумышленник разнесет его с иммобилайзером на расстояние большее, чем выставлено в настройках, связь прервалась. Более того, в случае нападения на пользователя тот может при помощи голосовой команды отключить систему.

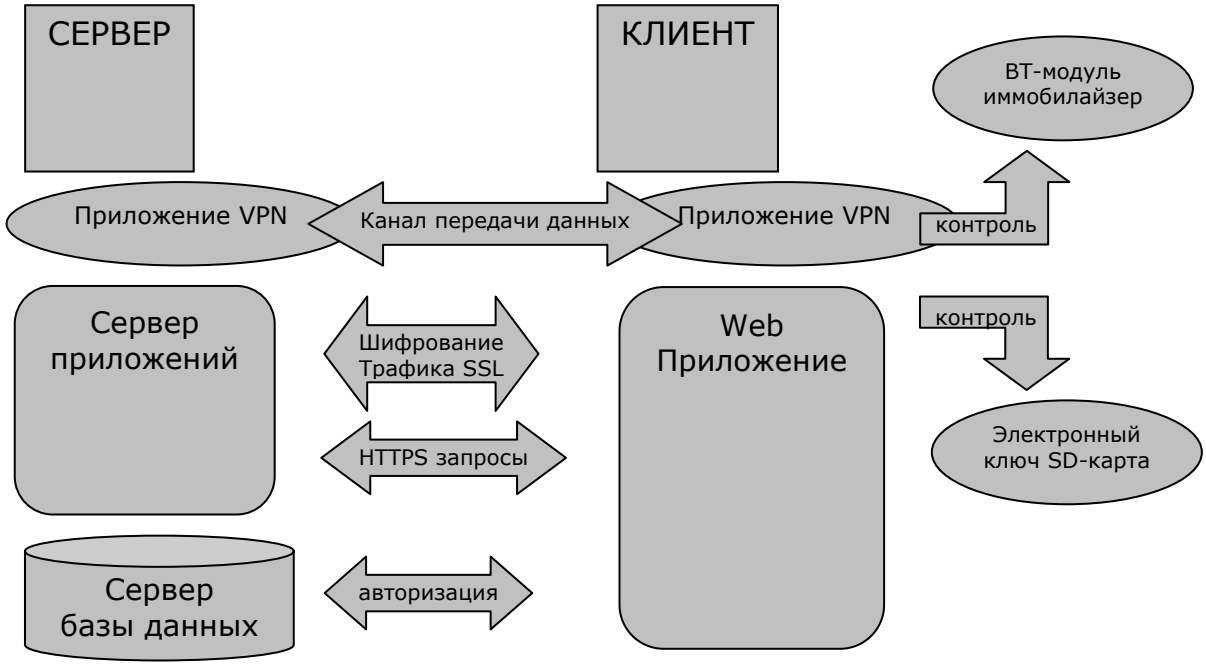



Рис. 2. Схема защиты автоматизированной системы управления предприятием

## Заключение

Успешная реализация данного проекта позволит решить многие задачи, с которыми приходится сталкиваться в процессе использования автоматизированной системы управления предприятием. Предложенные методы успешно сочетают в себе мобилизацию и повсеместную доступность получения сведений системы автоматизированного управления предприятием, а также обеспечивают должную защиту.

## Литература

1. Терьо М. и др. ORACLE Руководство по безопасности. М.: Лори, 2004. 576 с.
2. Кайт В.Б. Эффективное проектирование приложений Oracle. М.: Лори, 2006. 637 с.
3. Фридман А.Л. Построение Интернет-приложений на языке Java. М.: Горячая линия – Телеком, 2002. 336 с.

# МЕТОДЫ ФОРМИРОВАНИЯ ИНФОРМАЦИОННОГО ПОЛЯ В ИНФОРМАЦИОННОЙ СИСТЕМЕ

**Н.Ю. Дрюков**
**Научный руководитель – д.т.н., профессор Ю.А. Гатчин**

В работе описывается построение модели информационного поля, которое формируется любой информационной системой в процессе функционирования. Основная задача модели – провести частичную формализацию процесса утечки и перехвата информации и несанкционированных воздействий на нее.

## Введение

В настоящее время основным способом оценки защищенности информации и выявить каналы утечки является экспертный. Поиски некоей «формулы» защиты предпринимаются, но успешных способов структурировать весь комплекс угроз пока не найдено. В первую очередь это вызвано спецификой проблемы.

Проведение оценки информационной безопасности базируется на комплексе выявленных ранее угроз. Над обнаружением новых уязвимостей работают сразу два противоборствующих лагеря. Первый – злоумышленники, находящиеся в постоянном поиске «слабых мест» системы защиты. Второй – защитники, работающие для подготовки и применения превентивных мер. И если информация об угрозах, обнаруженных второй группой, общедоступна в большей своей части, то сведения об уязвимостях, выявленных первой группой, как правило, появляются после применения атаки на «слабое место», причем порой неоднократного.

Для любой информационной системы весь окружающий мир является враждебным, так как любой его объект несет в себе потенциальную угрозу безопасности. Более того, сами компоненты информационной системы являются источниками угроз, как правило, более опасных, чем внешние, хотя их количество более ограничено. Таким образом, в сфере защиты информации, как и в большинстве других областях обеспечения безопасности, сталкиваются стремящийся к бесконечности потенциал атакующего с ограниченными возможностями защищающегося.

Так как снизить количество «слабых мест», обнаруживаемых атакующим, не представляется возможным, то для решения этой проблемы следует попытаться повысить эффективность выявления уязвимостей защищающимся.

Применение экспертной оценки является эффективным средством в силу неформальности задачи по выявлению угроз, однако этот метод имеет ряд недостатков:

- невысокое количество хороших специалистов в этой сфере;
- расхождение мнений специалистов;
- оценка применимости только известных угроз;
- возможность ошибки, невнимательности, сложность охвата всего комплекса известных угроз.

Как отмечалось выше, основной причиной использования сугубо экспертной оценки защищенности является неформальность задачи. Таким образом, для решения данной проблемы необходимо формализовать объект оценки – каналы утечки информации. Для достижения этого результата попытаемся сформулировать концепцию комплексного информационного поля, представляющего собой единый канал утечки информации в информационной системе.

## Определение процесса утечки информации

Под информацией в статье понимаются любые сведения, возникающие и распространяющиеся в человеческом обществе.

Рассмотрим некую информационную систему, в которой циркулирует закрытая информация, предназначенная для ввода, хранения, обработки и уничтожения информации. В процессе функционирования информационной системы появляются каналы утечки защищаемой информации, которыми может воспользоваться злоумышленник. Как известно, канал утечки информации представляет собой источник информации, среду передачи и приемник информации. Таким образом, информация может распространяться только с помощью физических явлений, которые могут быть обнаружены и оценены человеком, в том числе и с помощью специальных инструментов.

Отображение информации – некое физическое явление, значения одного или нескольких параметров которого представляют собой закономерность, содержащую в себе данную информацию, т.е. физическое явление, содержащее в себе информативный сигнал. Следует заметить, что вся информация распространяется с помощью физических явлений. Информационное поле – совокупность отображений информации информационной системы. Информационное поле не несет единого отображения информации, даже когда в системе обрабатываются только информация с одним значением. Таким образом, информационное поле представляет собой множество отображений множества значений информации.

Основой для анализа информационного поля будет являться определение защищаемой информации и ее распределение по организации. Это позволит установить границы информационной системы.

После рассмотрения жизненного цикла информации и технологии его реализации становится возможным определить отображения – составляющие информационного поля. В результате проведения анализа отображений информации определяется возможность считывания каждого отображения.

В общем виде процесс утечки информации будет заключаться в выходе считываемого отображения информации за границы информационной системы.

Исходя из сказанного, можно сделать вывод о том, что отображение информации является исключительно источником проблем безопасности. Однако следует вспомнить, что основная цель любой информационной системы – автоматизация и упрощение обработки информации, что подразумевает использование ее отображения, так как по-другому невозможно передать работнику необходимые сведения. Таким образом, необходимо разделить все отображения на три группы: относящиеся к штатным функциям информационной системы (основные), не относящиеся к штатным функциям информационной системы (паразитные) и те, которые могут относится и первой группе и ко второй (неопределенные).

## Анализ отображения информации

Утечка информации может произойти, когда считываемое отображение покидает границы информационной системы. Опасность утечки будет зависеть от возможности считывания данного сигнала и от распространения этого сигнала относительно границ информационной системы.

При анализе конкретного отображения информации необходимо произвести следующие действия:

- определить параметры, заключающие в себе защищаемую информацию, способы считывания и применимые способы снижения считываемости;
- определить критичные для считывания сигнала параметры и применимые способы снижения считываемости;
- определить среду, пространство и принципы распространения отображения;

- определить критичные для распространения сигнала параметры, снижение которых наиболее эффективно снижает распространение, и применимые способы снижения распространения;
- определить максимально возможную критичность воздействия на информацию с учетом или без учета оценки важности информации.

Таким образом, для каждой точки пространства отображения информации и каждого отображения информации существует три параметра: считываемость ($R$), территориальность ($T$) и критичность ($C$).

Считываемость будет являться функцией от значения информативного параметра отображения для данной точки. Территориальность будет определяться возможностью применения средств считывания и напрямую будет зависеть от анализа системы охраны и контроля физического доступа в части применяемых технических средств с помощью концепции прерывистых окружностей (см. далее). Критичность будет определяться возможным воздействием на информацию (пассивным или активным, с определенным возможным уровнем доступа). Данные значения могут определяться экспертно, причем это возможно сделать однократно для каждого из возможных вариантов, которых существует ограниченное количество, и периодически пересматривать.

Оценить опасность отображения информации ($D$) можно с помощью формулы

$$D = \frac{R + T + C}{3}.$$

Следует отметить, что очень часто зависимость между параметрами одного физического явления нелинейна. Это приводит к тому, что свойства распространения и считывания отображения меняются при перемещении по диапазону значений критичных параметров. Поэтому необходимо сделать ранжирование диапазонов данных параметров и анализировать их как отдельные отображения.

### Распространение отображения информации

Пространство распространения отображения информации представляет собой совокупность источника отображения информации, среды распространения и ограничителей распространения отображения, являющихся границами распространения. Построение пространства отображения информации осуществляется соответственно степени важности защищаемых сведений. Основными принципами построения отображения будут следующие:

- отображение информации распространяется через соответствующую физическую среду, в том числе и по скрытым ее проявлениям;
- отображение информации может переходить в другое по форме и распространяться в другой среде, согласно законам физики, причем исходное отображение может продолжать распространяться.

Дополнительно при построении пространства отображения можно потребовать соблюдения еще двух принципов:

- любой объект, который не был проверен на декларированность всех его функций, является потенциально опасным.
- если значение отображения информации не уменьшилось до несчитываемых размеров в точке нахождения потенциально опасного объекта, в этой точке поле снова переходит в максимальное значение, а тело становится потенциальным источником.

К примеру, на рис. 2 расположено пространство распространения оптического отображения информации от монитора, а на рис. 3 – пространство отображения компьютерной информации, распространяющейся по вычислительной сети. На рис. 2 можно заметить несколько мест-неопределенностей, которые, в принципе, можно считать

уязвимостями в преградах, проводящими, а может быть, ретранслирующими отображение информации. В этих местах трудно оценить, имеет ли место утечка информации.

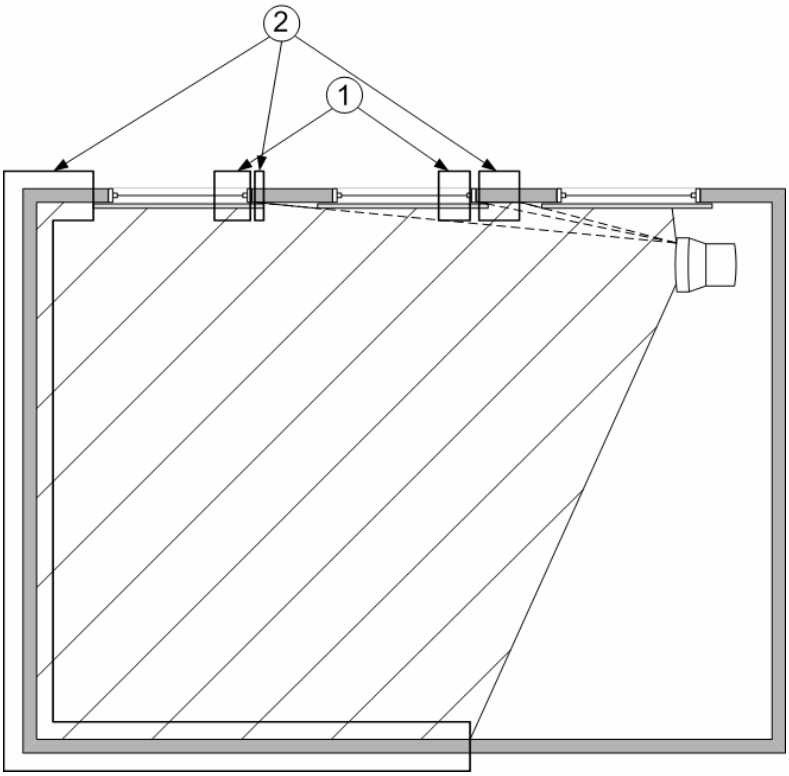



Рис. 2. Информационное поле монитора

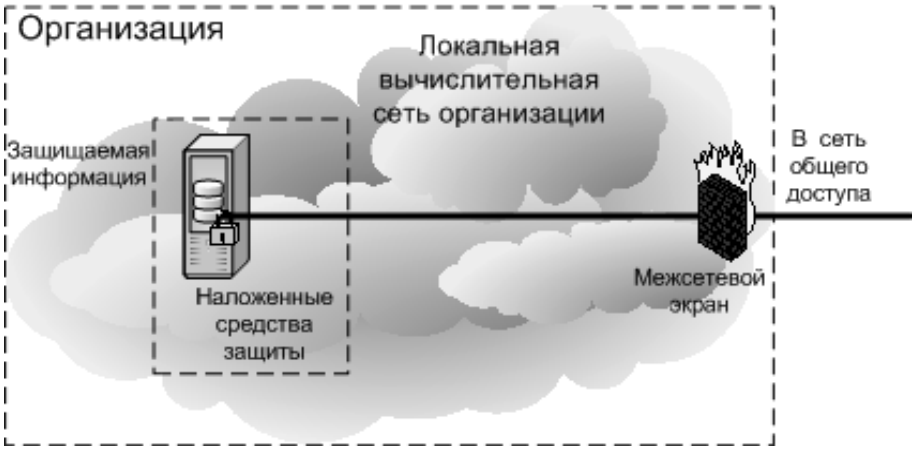



Рис. 3. Информационное поле компьютера в локальной вычислительной сети

Это связано с тем, что невозможно быть абсолютно уверенным в эффективности применения преград (см. указатель 1 на рис. 2) и отсутствии потенциальных источников отображения (см. указатель 1 на рис. 2), согласно четвертому принципу.

На рис. 3 пунктиром обозначены ограничения распространения отображения. В данном случае невозможно говорить о четких границах до проведения анализа применяемых защитных механизмов. Для этого их необходимо разделить на 2 группы. В первую группу войдут средства защиты, оказывающие влияние на считываемость (например, криптография), во вторую – средства, изменяющие территориальность (например, аутентификация и аудит). Для определения границ распространения следует провести анализ первой группы средств защиты. Особенностью определения считываемости таким путем будет отсутствие привязки к физическим явлениям и оценки исключительно сложности обхода данных механизмов.

Уточнение границ позволит определить «виртуальное» пространство распространения информационного поля со своими неопределенностями.

Возникает необходимость разрешить возникшие неопределенности, а также провести оценку отображения в тех местах, где нет неопределенности. Для этого можно применить концепцию прерывистых окружностей.

## Концепция прерывистых окружностей

Принцип распространения любого отображения информации можно описать с помощью концепции прерывистых окружностей (рис. 4). Данная концепция подразумевает тот факт, что распространение информационного поля ограничивается набором пар прерывистых окружностями, плотно прилегающих друг к другу. Каждая такая пара отображает одну подсистему защиты информации. Внутренняя окружность и «пробелы» в ней определяются характеристиками и надежностью конкретных технических средств, внешняя окружность и ее «пробелы» – действующей реализацией организационных мер.

Информационное поле ограничивается окружностями только в случае взаимного перекрытия всех пустот одной окружности линией другой. Другими словами, если рассмотреть информационную систему, в которую была внедрена высокоэффективная техническая защитная система с высокой степенью надежности и отличными характеристиками, то одно лишь это не дает гарантии защищенности информации, если не применяются организационные меры. То же можно сказать и о строгих организационных мероприятиях без применения технических средств защиты информации. Оба аспекта информационной безопасности должны закрывать «бреши» друг друга. Только в этом случае информационное поле теоретически не будет выходить за границы системы.

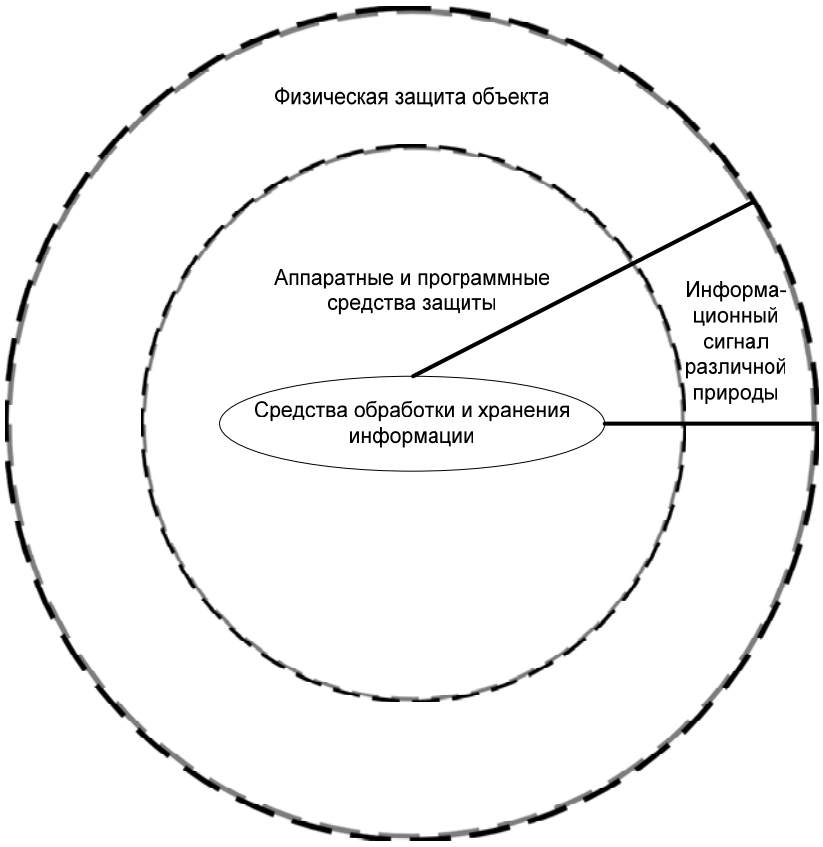



Рис. 4. Концепция прерывистых окружностей

Для оценки отображения необходимо ответить на вопрос: возможно ли злоумышленнику считать информационное поле до преград, и как ему это проще сделать?

Очевидно, что для считывания информационного поля злоумышленнику нужно самому находится на территории распространения информационного поля (в данном случае можно говорить и о «виртуальном» нахождении внутри локальной вычислительной сети). Для определения такой возможности необходимо определить возможные местоположения злоумышленника относительно информационной системы, фактические пути его проникновения на область со считываемым информационным полем (акцентироваться нужно на путях наименьшего сопротивления), преграды, которые возникают у него по мере продвижения и варианты их обхода, способы маскировки или «обоснования» своего нахождения на данной территории и возможность применения технических средств. Это позволит определить значение территориальности, применимое к большинству иных отображений информационной системы.

Для решения проблем неопределенности аналитик может:

- продолжить анализировать данное отображение информации для определения возможности утечки информации через эту уязвимость, расширяя таким образом пространство отображения (возможно, отображение не подлежат считыванию после утечки через эти уязвимости);
- предложить решения по избавлению от этих уязвимостей уже на данном этапе (тестовое решение – фактическая или аналитическая проверка возможности утечки, или перекрывающее решение – предложение по перекрытию потенциальной утечки);
- обосновать допустимость риска утечки информации через данную уязвимость (например, какая-либо сложно реализуемая уязвимость для некритичной информации).

Совокупность всех проанализированных отображений формирует единое информационное поле. Простое суммирование полученных значений оценки отображений в каждой точке позволит выявить места, где велика опасность утечки информации по нескольким каналам. Если посчитать среднее значение оценки в каждой точки, то можно определить места, где существуют наиболее критичные уязвимости.

## Заключение

В результате применения изложенной теории можно говорить о количественной оценке уязвимостей. Значения базируются на экспертной оценке, однако большая их часть может быть рассчитана однократно и применяться в большинстве информационных систем, периодически пересматриваясь в соответствии с техническим развитием средств съема информации.

## Литература


1. Шумский А.А., Шелупанов А.А. Системный анализ в защите информации. М.: Гелиос АРВ, 2005. 224 с.
2. Домарев В.В. Безопасность информационных технологий. Системный подход. Киев, 2004. 688 с.
3. ГОСТ 34.003-90. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения. М., 1990.
4. Галицкий А.В. и др. Защита информации в сети – анализ технологий и синтез решений // М.: ДМК Пресс, 2004. 616 с.
5. Каторин Ю.Ф., Куренков Е.В., Лысов А.В., Остапенко А.Н. Большая энциклопедия промышленного шпионажа / Под общ. ред. Е.В. Куренкова. СПб: Полигон, 2000. 511 с.

# ОСНОВНЫЕ НАПРАВЛЕНИЯ ОБЕСПЕЧЕНИЯ КОМПЛЕКСНОЙ ЗАЩИТЫ ИНФОРМАЦИИ КРУПНЫХ ПРЕДПРИЯТИЙ

**А.В. Годырева, Т.С. Николаева**
**Научный руководитель – к.т.н., доцент Н.С. Кармановский**

С переходом на рыночные отношения в условиях хозяйственной самостоятельности перед предприятиями возникают серьезные проблемы по обеспечению сохранности информации и безопасности. Само понятие «безопасность» включает в себя: информационную, коммерческую, юридическую и физическую безопасность. В статье рассматриваются актуальные вопросы по обеспечению комплексной защиты информации для крупных предприятий.

## Особенности информационной защиты крупных объектов

С помощью одного или нескольких технических средств либо только силами охраны практически невозможно обеспечить полную сохранность крупного объекта, занимающего значительную территорию и имеющего на балансе целый комплекс многоэтажных зданий. С развитием в стране различных форм собственности резко увеличилось количество предприятий, возросла конкурентная борьба между ними. Информация все более превращается в товар, а защита ее становится важнейшим направлением в деятельности фирм.

Увеличение числа предприятий часто сопровождается их «дроблением» на более мелкие хозяйственные объекты, численность работников в которых не превышает нескольких человек. Наряду с этим сохранились достаточно крупные предприятия, охрана которых зачастую не учитывает появление новых технических средств, а защита информации ограничивается упорядочиванием бумажного делопроизводства. Кроме того, даже государственные предприятия имеют коммерческую деятельность, в то время как законодательно коммерческая тайна определена сравнительно недавно.

В работе рассматриваются вопросы комплексной защиты информации крупных объектов. Интерес к крупным предприятиям обусловлен рядом причин:

- большой площадью предприятий;
- расположением предприятий в черте города, часто в зоне исторически сложившейся застройки;
- появлением новых форм собственности;
- расширением средств коммуникации и способов представления и передачи информации;
- сложностью создания единой системы защиты территории;
- большой протяженностью сетей коммуникаций;
- высокой стоимостью оборудования средств защиты, средств их интеграции, что доступно только крупным предприятиям.

Для исключения или существенного снижения возможных потерь материальных средств и информационных ресурсов служба безопасности объекта должна быть оснащена современным комплексом технических средств, что делает разработку системы контроля и управления доступом для режимного объекта своевременной и актуальной. Для эффективного решения этих задач необходим тщательный анализ существующих систем защиты, возможных способов несанкционированного доступа на объект, разработка новейших систем контроля и управления доступом, вспомогательного оборудования, что позволяет принять меры для противодействия возможным угрозам. Важнейшей задачей является своевременность обнаружения попыток несанкционированного проникновения на объект, несанкционированного доступа к конфиденциальной информации, а также принятия адекватных мер по ее нейтрализации.

**Интегрирование средств безопасности**

В целях повышения уровня безопасности крупных хозяйствующих объектов в настоящее время представляется целесообразным внедрение в структуру подобных предприятий интегрированных систем обеспечения безопасности (ИСОБ) − комплексных структур, предназначенных для обеспечения защищенного состояния наиболее важных объектов) (рис. 1) [1].


Рис. 1. Структура интегрированной системы обеспечения безопасности

Интегрированные системы безопасности представляют широкие возможности для объединения оборудования сторонних производителей, позволяют создать в единый комплекс безопасности системы видеонаблюдения, контроля доступа, видеоидентификации, охранно-пожарной сигнализации, мониторинга тревог, управления персоналом и посетителями. На сегодняшней день на рынке товаров и услуг предлагается широкий выбор интегрированных систем безопасности как зарубежных, так и отечественных производителей. Интегрированная система обеспечения безопасности должна удовлетворять следующим требованиям:

- тактическая надежность – выполнение системой безопасности своих функций при попытках вывода ее из строя или обхода охраняемых зон, предпринимаемых нарушителем,
- эксплуатационная надежность – безотказность оборудования при непрерывной работе в течение многих лет путем дублирования и резервирования основных систем, обеспечения централизованного и децентрализованного управления, обязательным обеспечения ключевых элементов средствами автономного питания.

Эффективное функционирование ИСОБ обеспечивают центральные посты, осуществляющие постоянный контроль и управление всеми техническими подсистемами. Аппаратура центрального поста должна обеспечивать максимальную автоматизацию текущих процедур в режиме нормального функционирования. В случае возникновения тревоги должно, по возможности, моментально привлекаться внимание персонала службы безопасности, а также обеспечиваться объективное и оперативное отображение и регистрация ситуации. Достигнутый уровень интеграции подсистем, входящих в ИСОБ, позволяет на единой программно-аппаратной платформе и с общей базой дан-

ных осуществлять отображение информации, управление и контроль состояния всех подсистем [2].

Современная концепция защиты объектов техническими средствами охраны (ТСО) заключается в выполнении четырех принципов:

- определение целей (предметов) защиты (кого и что защищать?),
- определение и оценка угроз (от кого защищать?),
- разработка и реализация адекватных мер защиты (как защищать?),
- создание и обеспечение функционирования комплексной защиты (рис. 2).

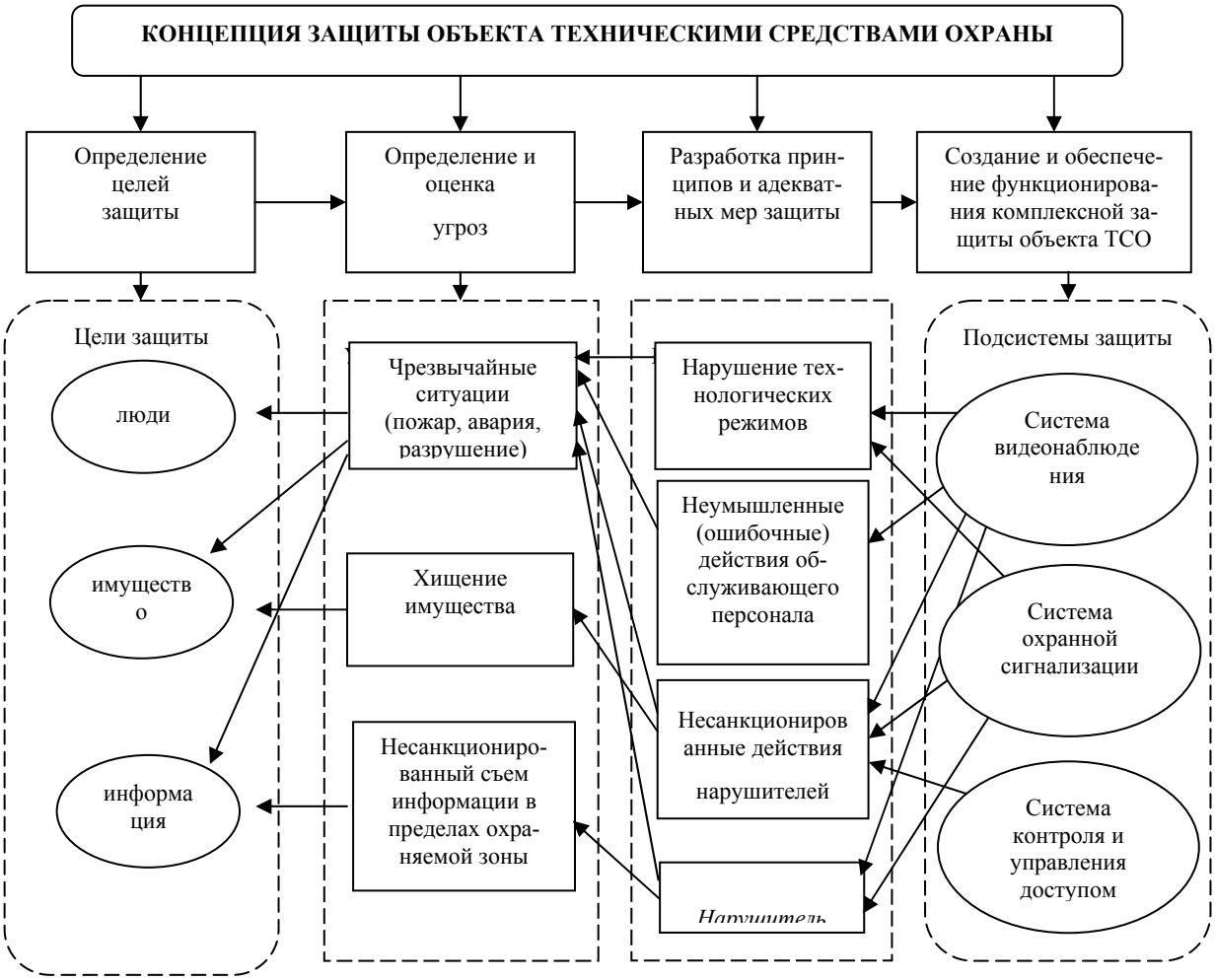



Рис. 2. Концепция защиты объекта техническими средствами охраны

**Принципы организации охраны крупного объекта**

На охраняемом предприятии можно выделить три группы объектов защиты: люди, имущество, информация. Цели защиты образуют пространство угроз, которые принято делить на естественные и искусственные. Система охраны должна выполнять следующие задачи:

- своевременно обнаружить факт совершения криминальной акции;
- транслировать сообщения о ней на центральный пункт охраны;
- провести верификацию (подтверждение истинности) тревожной ситуации;
- выдать команду тревожной группе для принятия адекватных мер по предотвращению ущерба;
- зарегистрировать факт возникновения любой аномальной ситуации, а также все действия оператора и персонала охраны для последующего анализа;

- нейтрализовать нарушителя.

Исходя из этих задач, система охраны должна включать в себя следующие основные составляющие:

- технические средства охраны − средства контроля, обнаружения и слежения,
- инженерные средства охраны − средства физической преграды,
- силы (подразделения) охраны − людские ресурсы.

Перечислим принципы построения систем охраны.

1. Системный подход к построению охраны техническими средствами. Данный принцип реализуется при выполнении следующих этапов:
   - обследование существующей системы ТСО,
   - выявление спектра угроз (составление описательной модели объектов охраны и нарушителя),
   - формирование концепции построения ТСО,
   - оценка возможностей сил охраны.
2. Обеспечение максимального эффекта при использовании ТСО.
3. Совместимость и развиваемость средств ТСО.
4. Многорубежность (многозональность) комплекса инженерно-технических средств охраны (КИТСО), в частности, для объектов особой важности.
5. Непрерывность рубежей ТСО.
6. Знание принципов построения КИТСО не должно позволить нарушителю «обойти» рубежи охраны и получить несанкционированный доступ на объект.

Задачи, выполняемые ТСО, порождают разнообразие типов технических средств, сложность устройства которых определяется конкретными условиями охраны. Тактико-технические требования, предъявляемые к ТСО, весьма разнообразны. Они делятся на три группы и должны учитывать условия эксплуатации монтажа, транспортировки, ремонта и безопасности:

- общие требования, предъявляемые ко всем видам ТСО;
- специальные тактико-технические требования, предъявляемые к отдельным видам средств обнаружения;
- частные требования, предъявляемые к конкретным типам аппаратуры.

### Выделение зон и формирование рубежей безопасности крупных предприятий

Равноценная защита крупных объектов представляется либо невозможной технически, либо экономически нецелесообразной. По этой причине актуально деление предприятия на зоны и рубежи безопасности, которые предусматривают различные уровни защиты. Рассмотрим предприятие ООО «Информационные системы» и на его примере определим вопросы по обеспечению комплексной безопасности крупных объектов. Данная организация занимается поставкой и установкой компьютерной техники для государственных и коммерческих организаций. Она находится в центре города и занимает площадь 5000 м$^2$, численность персонала составляет 250 человек.

Одним из принципов защиты информации на данном предприятии является выделение рубежей и создание зон безопасностей. Под зоной понимается территория, имеющая собственное ограждение и ограничение допуска лиц на эту территорию. Правильное формирование зон безопасности может не только повысить эффективность системы защиты, но и сократить расходы на ее оборудование. Типовыми зонами являются:

- территория, занимаемая учреждением и ограничиваемая забором или условной внешней границей;
- здание на территории;
- коридор или его часть;

- помещение (служебное, кабинет, склад и др.);
- шкаф, сейф, хранилище, в которых хранятся носители информации.

Рубежи охраны создаются на границе зоны с целью воспрепятствования проникновения на нее нарушителя (рис. 3).

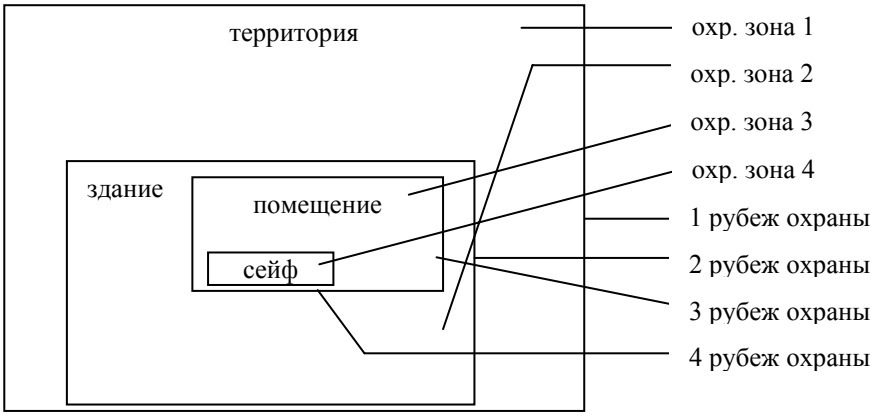



Рис. 3. Рубежи охраны и охраняемые зоны на пути движения нарушителя

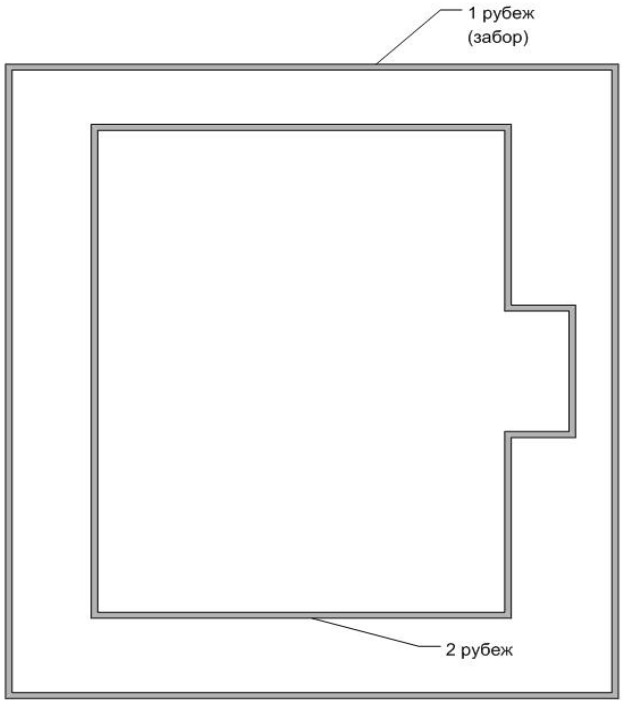



Рис. 4. План территории с выделением рубежей и зон безопасности

Особенностью рубежа охраны является равная прочность границ рубежей и наличие контрольно-пропускных пунктов или постов, обеспечивающих управление доступом в зону людей и автотранспорта. Чем большей ценностью обладает объект, тем большим количеством рубежей целесообразно окружать его источник. Если безопасность в каждой зоне обеспечивается только рубежом на ее границе, то для доступа нарушителя, например, к документу, хранящемуся в сейфе, ему необходимо преодолеть 4 рубежа: границу территории (забор), войти в здание, в помещение, открыть сейф [1].

На рис. 4 представлен план территории рассматриваемого предприятия.

Первый рубеж безопасности − периметр территории: пограничная часть территории объекта, которая просматривается сотрудниками охраны, не выходя из здания (например, с помощью средств видеонаблюдения). Для этого рассчитывается количество видеокамер, необходимых для просмотра 1-й зоны, установка возможных сигнальных

устройств, средств освещения и т.д. Особое внимание уделяется подходам к охраняемому объекту, подъездным путям, аварийным выходам и другим местам, способствующим подходу (подъезду) посторонних лиц к объекту в любое время суток.

Второй рубеж безопасности – периметр здания предприятия. Специалисту по охране необходимо четко уяснить размеры, площадь и конфигурацию охраняемого объекта. Он должен понять, какие технические мероприятия по защите необходимо осуществить, чтобы не допустить проникновения на объект криминальных элементов.

Внутри объекта также предполагается возможное выделение зон безопасностей в зависимости от степени важности циркулирующей информации и от степени защищенности помещения.

## Руководство безопасностью объекта

Когда речь заходит о безопасности предприятия, его руководство часто недооценивает важность информационной защиты. Основной упор часто делается на физическую безопасность (пропускной режим, охрану, систему видеонаблюдения и т.д.). Однако за последние годы ситуация существенно изменилась. Чтобы проникнуть в тайны предприятия, достаточно проникнуть в информационную систему или вывести из строя какой-либо узел компьютерной сети. Все это приведет к огромному ущербу, причем не только к прямому ущербу, который может выражаться в денежном эквиваленте, но и к косвенному. Для защиты секретов на предприятиях организуют службу безопасности (рис. 5), созданию которой обычно предшествует два события – это острое желание руководителей объекта отреагировать на внезапно возникшие реальные угрозы имуществу либо основанный на результатах исследований вывод о неудовлетворительном состоянии безопасности объекта. После детального изучения состояния безопасности предприятия появляется реальное представление о его системе безопасности, позволяющее осознано и целенаправленно проводить работу по обеспечению безопасности деятельности и самого объекта всеми его подразделениями и сотрудниками. Важной предпосылкой создания службы безопасности предприятия является разработка ее структуры, состава, положений о подразделениях и должностных инструкций для руководящего состава и сотрудников. Создавать ее надо осознанно и рационально, максимально используя опыт специалистов в сфере безопасности. Понимание своей роли и места в системе безопасности предприятия приведет ее к положительным результатам.

Надежность защиты информации, прежде всего, будет определяться тем, насколько правильно выбрана структура подразделения, на которое возложены эти функции, четкостью, с которой оно придерживается выбранного на предприятии режима конфиденциальности и безопасности. Предложенный вариант службы безопасности не претендует на бесспорность, однако призван помочь специалистам в вопросах организации работы по обеспечению безопасности защищаемого объекта.

Создание систем защиты информации обычно осуществляется в следующей последовательности:

- изучение объекта и обоснование необходимости создания системы,
- разработка проекта системы,
- приобретение необходимых средств,
- монтаж и оборудование системы,
- испытания и приемка системы [3].

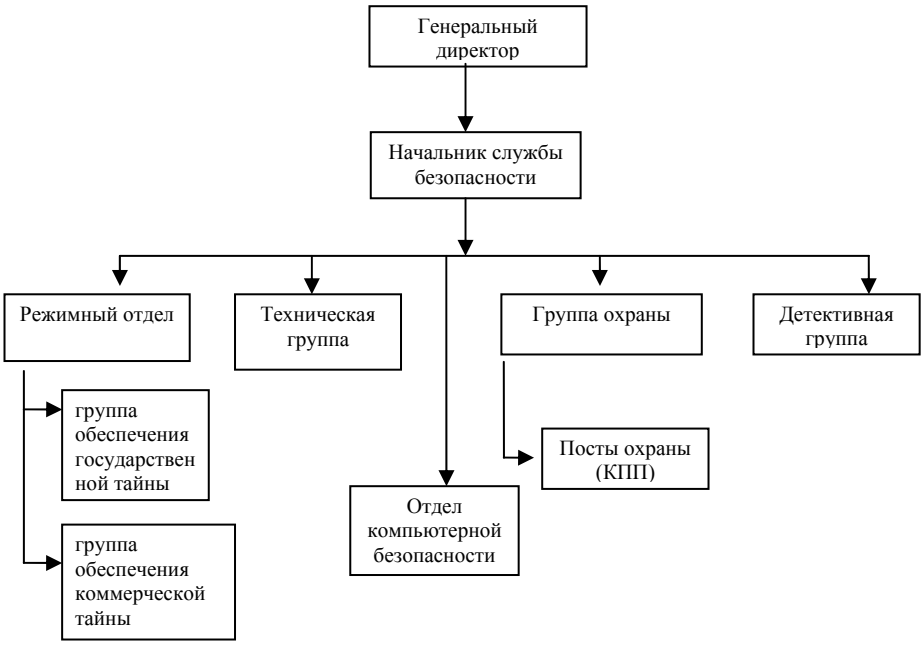



Рис. 5. Структура службы безопасности

## Заключение

Решение проблемы безопасности обеспечивается системным подходом к ней. Из практики применения системного анализа следует, что 50 % успеха в решении сложной задачи – ее правильная постановка. Чем более четко определены источники защищаемой информации, места их нахождения, способы и средства добывания информации злоумышленниками, тем конкретнее могут быть сформулированы задачи по защите и требования к соответствующим средствам. Источники информации определяются в результате структурирования защищаемой информации. Рост числа и видов угроз безопасности информации, сопровождающих повышение значимости информации в жизни общества и человека, представляют собой тенденцию, которую нельзя не учитывать. Не менее ответственные и сложные задачи возникают при непосредственном выборе рациональных способов и средств защиты, которые обеспечивают требуемый уровень защиты при минимальных затратах, не превышающих ущерб от хищения информации. В нахождении рациональных вариантов, удовлетворяющих этим условиям, состоит основная задача подразделений, занимающихся обеспечением защиты информации на предприятии.

Таким образом, сформулированы основные направления обеспечения комплексной защиты информации крупных объектов и предложены меры по повышению эффективности защиты:

- введение на предприятии интегрированной системы безопасности,
- формирование зон безопасности,
- создание службы безопасности.

## Литература

1. Андрианов В.И., Соколов А.В. Охранные системы для дома и офиса. СПб: БХВ-Петербург; Арлит, 2002. 304 с.
2. Максимов Ю.Н., Сонников В.Г. и др. Шпионские штучки. Технические методы и средства защиты информации. СПб: Полигон, 2000. 320 с.
3. Струков В.И. Экономика защиты информации. ТРТУ, 2000. 185 с.

# ГРАФИЧЕСКИЕ СТЕГОКОНТЕЙНЕРЫ

**С.С. Кувшинов, Н.Н. Прохожев**
**Научный руководитель – О.В. Михайличенко**

В работе предлагаются подходы к реализации задачи скрытой передачи данных с помощью неподвижных изображений. Эти подходы базируются на использовании особенностей системы человеческого зрения и избыточности визуальной информации. Освещены вопросы встраивания сообщений в графические стего-контейнеры, а также проблема выбора конкретного контейнера для того или иного объема информации. Также приводится пример одного из алгоритмов встраивания.

## Введение

Как известно, цель криптографии состоит в сокрытии содержания секретных сообщений. Стеганография идет принципиально дальше. Ее задача – скрыть от непосвященных сам факт существования сообщений. Такие скрытые сообщения могут включаться в различные внешне безобидные данные, вместе с ними храниться и передаваться без всяких подозрений со стороны. Эти данные – контейнеры стеганографических сообщений – могут иметь разнообразную природу. Для цифровой стеганографии это, прежде всего, файлы различных мультимедийных форматов (форматов изображений, музыкальных форматов, форматов видео и т.д.).

Возможность встраивания информации в изображения основана на факте некоторой избыточности визуальной информации. Цифровые фотографии (как и цифровая музыка, цифровое видео) представляются матрицами чисел, которые кодируют интенсивность сигналов в дискретные моменты в пространстве и/или времени. В нашем случае, контейнер (изображение) – матрица чисел, представляющих интенсивность света в определенный момент времени. Младшие разряды цифровых отсчетов содержат очень мало полезной информации. Их заполнение дополнительной информацией практически не влияет на качество восприятия изображения человеческим глазом, что и дает возможность скрытия в графическом файле нужной нам информации. С другой стороны, некоторые форматы изображений (JPEG) можно рассматривать как структуры, в которых для уменьшения объема информации для хранения точек используются зависимости, корреляции между близко расположенными друг к другу областями изображения.

Возникает два пути, по каждому из которых можно пойти при решении рассматриваемой задачи. Цель работы – рассмотрев особенности этих принципов, показать обоснованности применимости того или иного решения в конкретной ситуации.

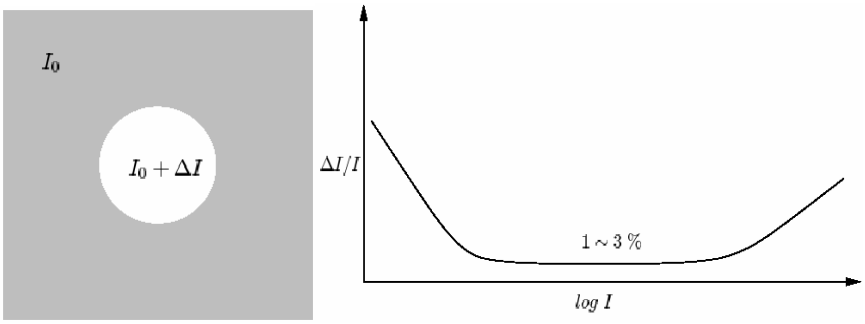
## Система человеческого зрения

Свойства системы человеческого зрения (СЧЗ) можно разделить на две группы: низкоуровневые («физиологические») и высокоуровневые («психофизиологические»). Выделяют три наиболее важных низкоуровневых свойства, влияющих на заметность постороннего шума в изображении:

- чувствительность к изменению яркости изображения;
- частотная чувствительность;
- эффект маскирования.

Чувствительность к изменению яркости можно определить следующим образом [1]. Испытуемому показывают некоторую однотонную картинку (рис. 1а). После того, как глаз адаптировался к ее освещенности $I$, «настроился на нее», постепенно изменяют яркость вокруг центрального пятна. Изменение освещенности $\Delta I$ продолжают до тех пор, пока оно не будет обнаружено. На рис. 1б показана зависимость минимального контраста $\Delta I / I$ от яркости $I$ (для удобства изменено привычное расположение осей). Как видно из рисунка, для среднего диапазона яркости контраст примерно постоянен,

тогда как для малых и больших яркостей значение порога неразличимости возрастает. Было установлено, что $\Delta I \approx 0.01 - 0.03I$ для средних значений яркости. Результаты исследований показывают, что при малых значениях яркости СЧЗ порог неразличимости уменьшается, т.е. СЧЗ более чувствительна к шуму в этом диапазоне.




а б

Рис. 1. Чувствительность к контрасту и порог неразличимости $\Delta I$

Частотная чувствительность СЧЗ проявляется в том, что человек гораздо более восприимчив к низкочастотному (НЧ), чем к высокочастотному (ВЧ) шуму. Можно показать, что частотная чувствительность тесно связана с яркостной. Известно также и выражение для определения порога маскирования на основе известной яркостной чувствительности, что позволяет найти метрику искажения изображения, учитывающую свойства СЧЗ. Такого типа математические модели хорошо разработаны для случая квантования коэффициентов дискретного косинусного преобразования изображения, так как именно оно применяется в стандарте JPEG.

Эффект маскирования в пространственной области может быть объяснен путем построения стохастических моделей изображения. При этом изображение представляется в виде марковского случайного поля, распределение вероятностей которого подчиняется, например, обобщенному гауссовскому закону.

Таким образом, можно предложить следующую обобщенную схему внедрения данных в изображение, которую так или иначе используют многие алгоритмы встраивания информации:

- Выполнить фильтрацию изображения при помощи ориентированных полосовых фильтров. При этом получим распределение энергии по частотно-пространственным компонентам.
- Вычислить порог маскирования на основе знания локальной величины энергии.
- Масштабировать значение энергии внедряемого ЦВЗ в каждом компоненте так, чтобы оно было меньше порога маскирования.

Высокоуровневые свойства СЧЗ пока редко учитываются при построении стего-алгоритмов. Основными из этих свойств являются:

- чувствительность к контрасту (высококонтрастные участки изображения, перепады яркости обращают на себя значительное внимание);
- чувствительность к размеру (большие участки изображения «заметнее» меньших размером). Существует порог насыщения, когда дальнейшее увеличение размера не существенно;
- чувствительность к форме (длинные и тонкие объекты вызывают большее внимание, чем круглые и однородные);
- чувствительность к цвету (некоторые цвета, например красный, «заметнее» других) Эффект усиливается, если фон заднего плана отличается от цвета фигур на нем;
- чувствительность к местоположению (человек склонен в первую очередь рассматривать центр изображения);
- люди обычно внимательнее к изображениям переднего плана, чем заднего;

- если на изображении есть люди, в первую очередь человек обратит свое внимание на них (на фотографии человек обращает первоочередное внимание на лицо, глаза, рот, руки);
- чувствительность к внешним раздражителям (движение глаз наблюдателя зависит от конкретной обстановки, от полученных им перед просмотром или во время него инструкций, дополнительной информации).

### Встраивание сообщения в незначащие биты контейнера

Цифровые изображения представляют собой матрицу пикселов. Пиксель – это единичный элемент изображения, имеющий фиксированную разрядность двоичного представления. Например, пиксели полутонового изображения кодируются 8 битами (значения яркости изменяются от 0 до 255).

Младший значащий бит (LSB) изображения несет в себе меньше всего информации. Известно, что человек обычно не способен заметить изменение в этом бите. Фактически он является шумом, поэтому его можно использовать для встраивания информации. Таким образом, для полутонового изображения объем встраиваемых данных может составлять 1/8 объема контейнера. Например, в изображение размером 512×512 можно встроить 32 килобайта информации. Если модифицировать два младших бита (что также почти незаметно), то можно скрытно передать вдвое больший объем данных.

Достоинства рассматриваемого метода заключаются в его простоте и сравнительно большом объеме встраиваемых данных. Однако он имеет серьезные недостатки. Во-первых, скрытое сообщение легко разрушить. Во-вторых, не обеспечена секретность встраивания информации. Нарушителю точно известно местоположение всего ЦВЗ. Для преодоления последнего недостатка было предложено встраивать ЦВЗ не во все пикселы изображения, а лишь в некоторые из них, определяемые по псевдослучайному закону в соответствии с ключом, известным только законному пользователю. Пропускная способность при этом уменьшается. Рассмотрим подробнее вопрос выбора пикселов изображения для встраивания в них скрытого сообщения.

### Встраивание

Характер поведения младшего значащего бита изображений чаще всего неслучаен. Скрываемое сообщение не должно изменять статистики изображения. На практике обычно ограничиваются поиском пикселов, модификация которых не вносит заметных искажений в изображение. Затем из этих пикселов в соответствии с ключом выбираются те, которые будут модифицироваться. Скрываемое сообщение шифруется с применением другого ключа. Этот этап может быть дополнен предварительной компрессией для уменьшения объема сообщения.

Рассмотрим пример – обработка изображения в формате BMP [3]. Если взять картинку, лучше всего TrueColor в 24-х битном формате, и изменить по какому-то известному только нам принципу младшие значащие биты цвета, то на глаз разница между исходным и полученным изображениями не будет заметна. 24 бита – очень удобный формат хранения информации об изображении, при таком представлении за один цветовой канал отвечает отдельный байт. Внедрение может осуществляться так.

1. Берем сообщение и предварительно подготавливаем его – шифруем и пакуем. Этим достигаются сразу две цели – повышение КПД и увеличение стойкости системы. В начало для удобства можно записать сигнатуру метода, что не секретно, зато просто.
2. Берем контейнер и внедряем в его младшие биты подготовленное сообщение любым удобным для нас способом, например, раскладываем упакованное сообщение

в битовую последовательность и заменяем избыточные биты (НЗБ) контейнера битами сообщения.

Надежность такого внедрения прямо пропорциональна соответствию характера распределения НЗБ в контейнере и сообщении. А распределения эти в подавляющем большинстве случаев совпадать не будут. В некоторых случаях это будет визуально заметно на картинке, построенной из одних только младших битов контейнера. Например, берем и картинку и внедряем в нее информацию (см. рис. 2).

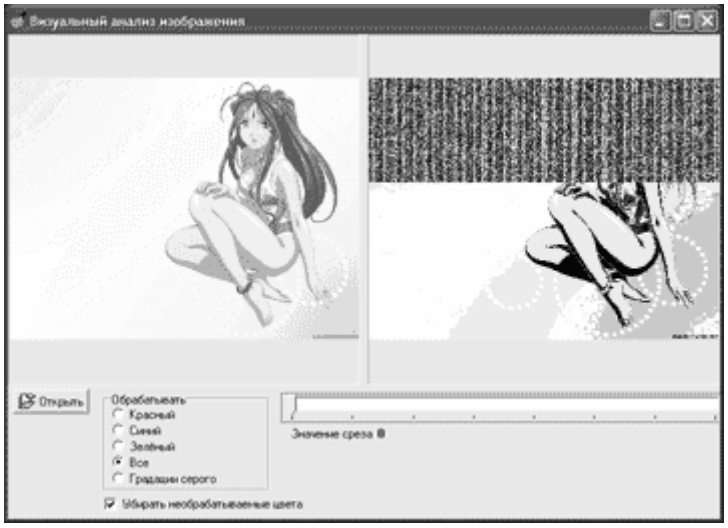



Рис. 2. Визуальный анализ изображения

При помощи простой утилиты, показывающей картинку побитно (т.е. только интересующие нас биты нужных цветовых компонентов) видно внедрение данных, т.е. файл внедрения легко определить. Конечно, для большинства наблюдателей достаточно и того, что «на глаз не видно», но слабые места своей защиты надо знать.

## Особенности BMP контейнеров

Все BMP контейнеры можно разделить на два класса: «чистые» и зашумленные [3, 5]. В «чистых» картинках прослеживается связь между младшим битом и остальными 7-ю битами элементов цвета, а также существенная зависимость самих младших битов между собой. Внедрение сообщения в «чистую» картинку разрушает существующие зависимости, что видно из рис. 3. Если же картинка зашумлена (например, получена со сканера или фотокамеры), то определить вложение становится на порядок сложнее. Различить, какой контейнер попал вам под руку, можно тоже побитным просмотром картинки. Например, сканированное изображение, в которое ничего не внедрялось, показано на рис. 3.

Объясняется это шумом матрицы сканера (или цифровой камеры). Следует сказать, что просмотром и анализом распределения НЗБ иногда очень просто определить факт компьютерной обработки (подчистки) картинки (рис. 4 – изображение без внедрения, рис. 5 – то же изображение, но с внедрением). Четко видна граница между «своим» шумом и шумом, возникшим в результате внедрения сообщения. Отсюда вытекает еще одно необходимое правило – всегда нужно распределять биты сообщения по всем младшим битам контейнера (например, внедрять не последовательно, а в каждый 2-й, 3-й и т.д. НЗБ) или же дополнять чем-то (шумом с тем же законом распределения) длину сообщения так, чтобы она стала равна объему НЗБ контейнера. Еще одно интересное решение – подбор картинки под сообщение: из множества имеющихся картинок подобрать такую, которая исказится менее всего при внедрении сообщения.

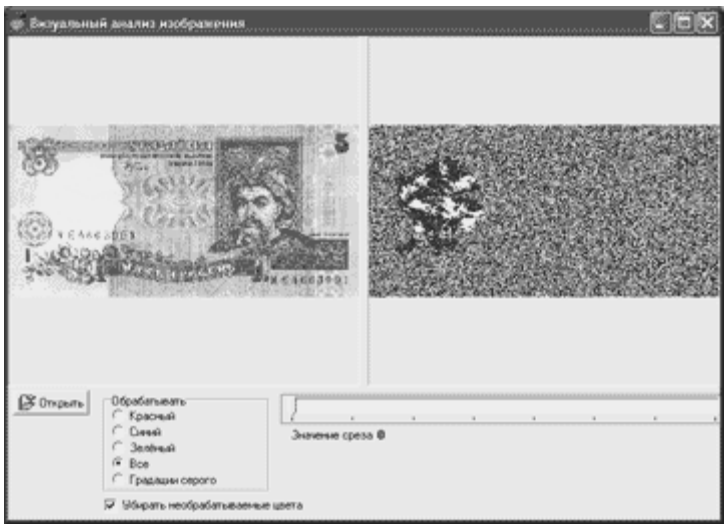



Рис. 3. Отсканированное изображение

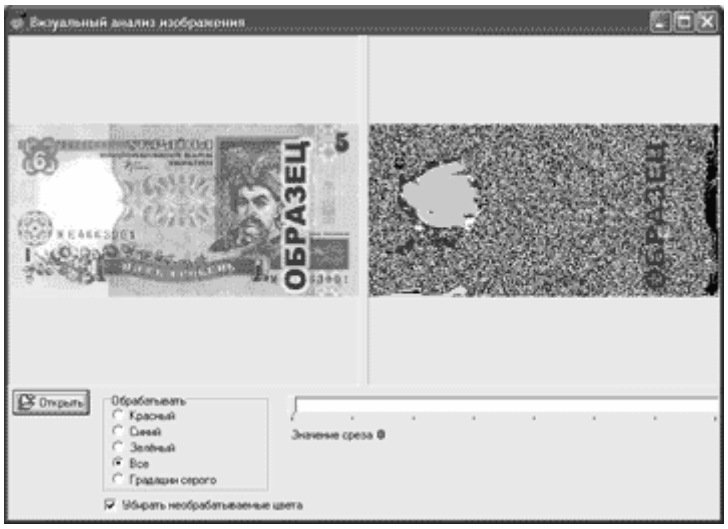



Рис. 4. Изображение без внедрения сообщения

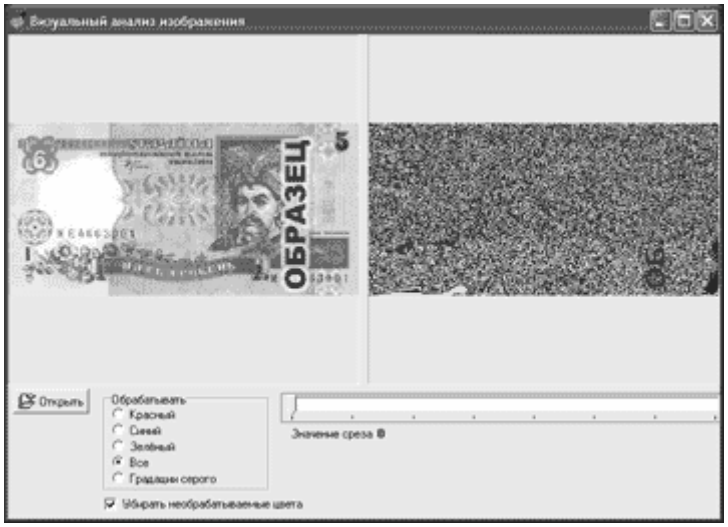



Рис. 5. Изображение с внедренным сообщением

## Ситуация с форматом JPEG

Другим популярным методом встраивания сообщений является использование особенностей форматов данных, использующих сжатие с потерей данных (например, JPEG). Этот метод (в отличие от LSB) более стоек к геометрическим преобразованиям и обнаружению канала передачи, так как имеется возможность в широком диапазоне варьировать качество сжатого изображения, что делает невозможным определение происхождения искажения.

Если в случае BMP нас интересовали младшие биты градаций серой шкалы или RGB-значений при цветном изображении, то в формате JPEG речь пойдет о другом виде информационной избыточности.

JPEG – формат, в котором для уменьшения объема информации для хранения точек используются зависимости, корреляции между близко расположенными друг к другу областями изображения [2].

Стандарт сжатия JPEG является в настоящее время наиболее распространенным и своеобразным «benchmark`ом» для алгоритмов цифровых водяных знаков (ЦВЗ) (т.е. устойчивость системы ЦВЗ к сжатию JPEG проверяется обычно в первую очередь). В соответствии с этим стандартом изображение разбивается первоначально на блоки 8×8 элементов, к каждому из которых применяется дискретное косинусное преобразование (ДКП). Назначением ДКП является перераспределение энергии: значимые коэффициенты группируются в левом верхнем углу квадрата спектральных коэффициентов, так как соседние пикселы изображения коррелированы. Далее следуют равномерное табличное квантование коэффициентов, кодирование длин серий и кодирование Хаффмана.

Вейвлет-преобразование, также как и ДКП, перераспределяет энергию изображения. Эта компактность энергии ведет к эффективному применению скалярных квантователей. Однако они не учитывают остаточную структуру, сохраняющуюся в вейвлет-коэффициентах, в особенности высокочастотных субполос. Современные алгоритмы сжатия тем или иным образом используют эту структуру для повышения эффективности сжатия.

## Алгоритм встраивания “F5”

Данный алгоритм встраивает биты сообщения в произвольным образом выбираемые коэффициенты ДКП [4]. Для его реализации необходимо определить:

1. качество стегопосылки;
2. изображение-контейнер;
3. выходной файл;
4. файл с сообщением;
5. пароль для встраивания.

Процесс встраивания состоит из 6 следующих шагов:

1. получение RGB-представления изображения-контейнера;
2. расчет таблицы квантования исходя из требования качества стегопосылки, сжатие изображения путем сохранения значений ДКП-коэффициентов;
3. преобразование изображения в сигнал с нулевым средним и определенной дисперсией так, чтобы абсолютные значения коэффициентов ДКП находились в диапазоне (200, 250);
4. определение распределения битов сообщения по коэффициентам с использованием указанного пароля;
5. разделение сообщения на сегменты по $k$ бит, которые встраиваются в небольшие по размеру блоки коэффициентов ДКП изображения. Этот процесс имеет то преимущество, что при этом существует возможность адаптации к локальной яркости и

гладкости изображения. Однако при достаточной энергии встраиваемых битов появляется эффект блочности;

6. оценка результата. Если длина сообщения укладывается в оцененную размерность, встраивание прошло успешно. В противном случае возникает ошибка переполнения контейнера.

## Заключение

Анализ тенденций развития компьютерной стеганографии показывает, что в ближайшие годы интерес к ней будет усиливаться все больше и больше. Сильнейшим катализатором этого процесса является лавинообразное развитие Internet, в том числе такие нерешенные проблемы, как защита авторского права, защита прав на личную тайну, организация электронной торговли, компьютерная преступность и кибертерроризм.

Очевидно, цифровые изображения будут еще очень долгое время применяться в качестве стегоконтейнеров, поскольку цифровые фотографии распространены повсеместно, и обмен файлами такого типа – весьма распространенное действие, чтоб вызывать подозрения факта скрытой передачи информации.

Наиболее перспективными являются алгоритмы, приспособленные для работы с файлами JPEG и JPEG 2000. Эти форматы допускают сжатие за счет избыточности визуальной информации, основанной на корреляции соседних областей изображения. В этом случае изображение рассматривается как сигнал, который может подвергаться преобразованиям Фурье, вейвлет-преобразованиям, дискретному конусному преобразованию.

## Литература


1. Воробьев В.И., Грибунин В.Г. Теория и практика вейвлет-преобразования. Учебное пособие. СПб.: ВУС, 1999. 204 с.
2. Грибунин В.Г. Цифровая стеганография. СПб.: Солон-Пресс, 2002. 272 с.
3. Стеганография. Особенности использования программ на основе метода наименьшего значащего бита, URL http://www.compdoc.ru/secur/cript/steganografiya/
4. Fridrich J., Goljan M., Hogea D. Steganalysis of JPEG Images: Breaking the F5 Algorithm http://www.ws.binghamton.edu/fridrich/Research/f5.pdf.
5. Основные положения стеганографии, URL http://www.citforum.ru/security/articles/stegano.shtml

# ОСОБЕННОСТИ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ ПРОМЫШЛЕННЫХ СИСТЕМ

**А.Л. Липатов**
**Научный руководитель – д.т.н., профессор, Ю.А. Гатчин**
**Э.В. Белов, М.В. Масленников**
**Научный руководитель – д.т.н., профессор, А.Г.Коробейников**

В статье производится анализ существующих проблем обеспечения информационной безопасности существующих автоматизированных систем управления и рассматриваются основные специфические виды угроз на данные системы.

## Введение

Автоматизированные системы управления (АСУ и АСУТП) в настоящее время используются в большинстве отраслей промышленности, в нефти и газодобыче, на электростанциях и железных дорогах, на пивоварнях и лыжных курортах. В мире эксплуатируются миллионы промышленных систем (ПС), стоимость которых измеряется тысячами и миллионами долларов США. Степень зависимости критической инфраструктуры государства от таких систем неуклонно возрастает, и вопросы обеспечения их информационной безопасности приобретают первостепенное значение.

В отличие от других видов автоматизированных информационных систем, ПС, особенно те, которые используются для управления критической инфраструктурой, имеют ряд особенностей, обусловленных их особым назначением, условиями эксплуатации, спецификой обрабатываемой в них информации и требованиями, предъявляемыми к функционированию. Главной же особенностью ПС является то, что с их помощью в автоматическом, либо полуавтоматическом, режиме в реальном времени осуществляется управление физическими процессами и системами, от которых непосредственным образом зависит наша безопасность и жизнедеятельность: электричество, связь, транспорт, финансы, системы жизнеобеспечения, атомное и химическое производство и т.п.

Обеспечение информационной безопасности ПС требует особого подхода, учитывающего эти особенности. Для того чтобы выработать такой подход, необходимо, прежде всего, оценить серьезность проблемы в целом, затем, опираясь на накопленную статистику инцидентов, подвергнуть тщательному анализу специфические для ПС угрозы и уязвимости и на основании этого анализа определить особые требования к режиму обеспечения информационной безопасности критической инфраструктуры.

## Угрозы безопасности промышленных систем

ПС эволюционировали от экзотических программных и аппаратных средств в 70-х до вполне современных систем, в которых используются стандартные IBM-совместимые ПК, операционные системы семейства Microsoft Windows, сетевые протоколы TCP/IP, Web-браузеры, доступ в Интернет. Множество угроз в отношении этих систем значительно расширилось благодаря такой стандартизации, а также благодаря распространенной практике подключения промышленных систем к ЛВС организации и использованию в них технологий беспроводного доступа.

Умышленные угрозы в отношении ПС, в зависимости от того, кто выступает в качестве источника угрозы, можно разделить на следующие основные группы.

1. Вредоносное ПО. ПС, как и любые другие ИТ системы, потенциально подвержены угрозам со стороны компьютерных вирусов, сетевых червей, троянских программ и программ шпионов.
2. Инсайдеры. Внутренние пользователи, хорошо знающие систему изнутри, как показывает практика, представляют собой одну из основных угроз. Инсайдер может

умышленно повредить оборудование или программное обеспечение. Администраторы и инженеры, обслуживающие систему, могут также неумышленно нанести вред ее функционированию, допустив ошибку в настройках системы или нарушение правил обеспечения безопасности.

3. Хакеры. Аутсайдеры могут быть заинтересованы в исследовании возможности получения доступа и контроля над системой, мониторинге трафика и реализации атак на отказ в обслуживании.
4. Террористы. Это наиболее серьезная угроза, создающая основные различия между системами, относящимися к критической инфраструктуре и обычными ИТ системами. Террористы заинтересованы в том, чтобы вывести систему из строя, нарушить процессы мониторинга и управления, либо получить контроль над системой и нанести как можно больший вред критической инфраструктуре.

Однако в критичных отраслях, преимущественно использующих ПС, отсутствуют два основных мотивирующих фактора для киберпреступности. Это экономические стимулы, к которым относятся кредитные карты и электронные счета, лежащие в основе многих компьютерных преступлений, и коммерческие тайны, являющиеся основной целью промышленного шпионажа.

## Кибератаки на промышленные системы

Существует большое количество зарегистрированных инцидентов безопасности, затрагивающих системы управления критической инфраструктурой. В ряде научно-исследовательских институтов, ФРБ и других организациях ведется соответствующая статистика. Согласно этой статистике, в США на ПС осуществляется не менее 100 кибератак в год и существует тенденция к непрерывному увеличению их числа. Зафиксированы все категории кибератак за исключением кибертерроризма.

Анализ текущей ситуации показывает, что хотя теоретически и существует возможность электронных вторжений в критичные системы управления, создающих серьезные, в том числе и физические, угрозы безопасности, получение контроля над такими системами извне является крайне маловероятным событием. В настоящее время реальность такова, что легче уничтожить цель путем физического воздействия, нежели поразить ее путем взлома компьютерной системы.

Кибератаки действительно могут иметь серьезные последствия, хотя и не связанные с нанесением ущерба жизни и здоровью людей, массовыми разрушениями и другими катастрофами. В наихудшем сценарии хорошо спланированная массированная кибератака может временно вывести из строя системы телекоммуникаций в густо населенных районах [1].

Ядерный завод в штате Огайо функционировал в автономном режиме в течении года после того, как сетевой червь SQL Slammer привел к отключению системы отображения периметра безопасности на пять часов и заводского компьютера, используемого для мониторинга производственного процесса, на шесть часов. Для обеих систем были предусмотрены дублирующие аналоговые системы, которые не пострадали. Заводская производственная сеть была непосредственно подключена к корпоративной сети, в которую «червь» проник по удаленному каналу из партнерской сети.

Показательным примером кибератаки со стороны инсайдера может послужить дело Вайтека Бодена, приговоренного к двум годам лишения свободы за причинение умышленного ущерба канализационной системе Совета австралийского графства Маручи. Боден работал контролером в компании, которая устанавливала данную систему, включавшую 150 насосных станций. Станции обменивались данными между собой и с центральным компьютером при помощи локальных процессоров. Когда проект был завершен, Боден уволился из компании и попытался устроиться на работу к бывшему за-

казчику, но получил отказ. После этого начались проблемы на насосных станциях. Постепенно стало ясно, что причина заключается не в системных сбоях. Система сигнализации отключалась, связь неожиданно обрывалась, насосы не включались в нужное время, – в результате происходил выброс нечистот. Боден проделывал все это из собственного автомобиля при помощи ноутбука, радиопередатчика и локального процессора, позаимствованного у бывшего работодателя.

Примером хакерской атаки на критическую инфраструктуру США может служить удаленный взлом в 2001 году компьютерной сети Независимого системного оператора Калифорнии, управляющего электросетью штата. Хотя тогда хакерам не удалось получить доступ к действующей системе управления электросетью, они имели доступ к корпоративной сети в течение 17 дней. Намерения хакеров и их происхождение так и остались невыясненными [2].

## Уязвимости промышленных систем

На начальном этапе развития в ПС использовалось малоизвестное специализированное аппаратное и программное обеспечение, а их сетевое взаимодействие с внешним миром было сильно ограничено. Круг возможных угроз был слишком узок, поэтому внимания вопросам информационной безопасности со стороны разработчиков и владельцев таких систем практически не уделялось. Со временем разработчики переходят на стандартные платформы, а владельцы ПС, с целью повышения эффективности управления, подключают их к смежным системам. Существующая тенденция к повышению открытости и стандартизации ПС повышает их уязвимость к кибератакам, однако среди экспертов не существует единого мнения относительно того, насколько сложной для аутсайдера задачей является получение доступа к ПС.

В системах критической инфраструктуры существуют те же самые уязвимости, что и в большинстве обычных ИТ систем. Кроме этого, особенности промышленных систем, обусловливают существование в них уникальных уязвимостей.

1. Человеческий фактор. Эксплуатацией промышленных и корпоративных систем обычно занимаются разные подразделения. Персонал ПС, как правило, достаточно далек от вопросов обеспечения информационной безопасности, в его составе нет соответствующих специалистов, а рекомендации ИТ персонала на него не распространяются. Основной задачей остается решение технологических проблем, возникающих в ходе эксплуатации системы, обеспечение ее надежности и доступности, повышение эффективности и минимизация накладных расходов.
2. Уязвимости операционных систем. Уязвимости операционных систем в равной степени свойственны и ПС, и корпоративным системам, однако установка обновлений для программной части в ПС на регулярной основе зачастую не выполняется. Главной заботой администратора такой системы является ее бесперебойная работа. Установка предварительно не протестированных программных коррекций может повлечь серьезные проблемы в функционировании, а на полноценное тестирование обычно нет ни времени, ни средств.
3. Слабая аутентификация. Использование общих паролей является обычной практикой для ПС. Благодаря этому у персонала пропадает ощущение подотчетности за свои действия. Системы двухфакторной аутентификации используются довольно редко, а ключевая информация зачастую передается по сети в открытом виде.
4. Удаленный доступ. Для управления ПС довольно часто используется удаленный доступ по коммутируемым каналам или по VPN каналам через сеть Интернет. Это может привести к серьезным проблемам с безопасностью.
5. Внешние сетевые подключения. Отсутствие соответствующей нормативной базы и соображения удобства использования порой приводят к тому, что между ПС и кор-

поративными системами создаются сетевые подключения. Можно услышать даже рекомендации по поводу использования «комбинированных» сетей, позволяющих упростить администрирование и улучшить безопасность.

6. Средства защиты и мониторинга. В отличие от корпоративных систем, использование систем определения вторжения, брандмауэров и антивирусов в ПС не является распространенной практикой, а для анализа журналов аудита безопасности обычно не остается времени.
7. Беспроводные сети. В ПС часто используются различные виды беспроводной связи, включая протоколы 802.11, без использования достаточных возможностей по защите.
8. Удаленные процессоры. Определенные классы удаленных процессоров, используемых в ПС для контроля технологических процессов, содержат известные уязвимости. Производительность этих процессоров не всегда позволяет реализовать функции безопасности. Кроме того, после установки их стараются не трогать годами, на протяжении которых они остаются уязвимыми.
9. Программное обеспечение. Программное обеспечение ПС обычно не содержит достаточного количества функций безопасности. Кроме того, оно не лишено архитектурных слабостей.
10. Раскрытие информации. Нередко владельцы ПС сознательно публикуют информацию об их архитектуре. Консультанты и разработчики частенько делятся опытом и раскрывают информацию о бывших клиентах.
11. Физическая безопасность. Удаленные процессоры и оборудование ПС могут находиться за пределами контролируемой зоны. В таких условиях они не могут физически контролироваться персоналом, и единственным механизмом физической защиты становится использование железных замков и дверей, а такие меры уж точно не являются серьезным препятствием для террористов.

Таким образом, существует значительное количество уязвимостей, являющихся специфичными для ПС. Эти уязвимости обуславливают особые требования по безопасности и особые режимы эксплуатации таких систем.

### Мероприятия по защите промышленных систем

Формирование взглядов и приоритетов в области обеспечения информационной безопасности ПС критической инфраструктуры хорошо прослеживается на примере США – страны, в наибольшей степени подверженной угрозам кибертерроризма. В 1997 году на свет появился отчет Президентской комиссии по защите критической инфраструктуры, который послужил точкой отсчета для начала широкомасштабных действий, предпринимаемых правительством США с целью повышения защищенности физической, сетевой и информационной инфраструктуры государства. В 1998 году указ Президента США №63 (PDD63) определил понятие критической инфраструктуры как «физические и информационные системы, необходимые для обеспечения минимально допустимого уровня функционирования экономики и правительства». К критической инфраструктуре были отнесены: телекоммуникации, энергетика, банковская и финансовая система, транспорт, водные системы и аварийные службы. PDD63 также определил основные элементы государственной стратегии в области защиты критической инфраструктуры, к числу которых относятся:

- важность сотрудничества между общественным и частным сектором;
- головные федеральные агентства для каждого сектора критической инфраструктуры;
- координационные группы для координации усилий федеральных агентств и промышленных групп;

- система оповещения и обмена информацией в рамках Национального центра защиты инфраструктуры (NIPC);
- системы обмена информацией для каждого сектора промышленности, известные как Центры сбора и анализа информации (ISAC);
- требование создания «Плана обеспечения безопасности национальной инфраструктуры», устанавливающего контрольные точки для анализа уязвимостей и подготовки планов их ликвидации в каждом секторе промышленности.

В 2001 году после известных событий 11 сентября был выпущен «Акт о защите критической инфраструктуры», а в 2002 году – «Акт о безопасности Отечества», в соответствии с которым в США был образован Департамент национальной безопасности (DHS) и учреждена должность Директора по анализу информации и защите инфраструктуры. В 2003 году Президентом США была утверждена «Национальная стратегия обеспечения безопасности киберпространстра» (5). Этот объемный документ адресован широкой американской общественности и направлен на расширение взаимодействия и консолидацию усилий различных слоев общества, государственных, общественных и частных организаций в деле противодействия кибертерроризму. Основная часть Стратегии расставляет приоритеты по созданию системы ответных мер, программы противодействия угрозам и уязвимостям, программы обучения и повышения осведомленности, национальной и международной кооперации. Повышение защищенности ПС было объявлено национальным приоритетом.

С этого времени в мероприятиях по обеспечению безопасности ПС задействованы разработчики и владельцы этих систем, консультанты и научно-исследовательские организации, независимые ассоциации и государственные учреждения. Для отработки технических решений по защите ПС были созданы тестовые лаборатории (например, National SCADA Test Bed). В результате было выпущено и продолжается разработка значительного количества стандартов и руководств по различным аспектам обеспечения безопасности ПС.

Компанией Digital Bond Inc. был разработан «Профиль защиты центра управления для ПС управления», в котором формализуется перечень из 22 видов угроз в отношении центра управления ПС, на основании данного перечня формулируются 28 задач защиты, исходя из которых определяется 55 компонентов функциональных требований безопасности и 17 компонентов требований к гарантированности оценки в соответствии с Общими Критериями (4). Этой компанией был также разработан набор сигнатур сетевых атак для протоколов Modbus TCP и DNP3, используемых в ПС. Эти сигнатуры изначально были разработаны для системы Snort в рамках исследовательского проекта, финансируемого Департаментом национальной безопасности США. Поддержка сигнатур была добавлена в соответствующие продукты Symantec и ISS, являющихся лидерами в этом сегменте рынка информационной безопасности. Компания Cisco модифицировала эти сигнатуры для работы на своей платформе (они были включены в S198 Signature Update для Cisco IDS версии 4.1 и IPS версии 5.0). Продолжается исследовательская работа по созданию механизмов защиты и для других распространенных сетевых протоколов, используемых в ПС.

В ходе планирования и реализации организационно-технических мер по защите ПС необходимо, прежде всего, опираться на международные стандарты ИБ, наиболее востребованным из которых в настоящее время является ISO/IEC 17799:2005. Описанные в этом стандарте механизмы контроля в полной мере применимы и к ПС. Учитывая повышенные требования по защите критической инфраструктуры, в дополнение к существующим международным стандартам, определяющим только базовые механизмы безопасности, необходимо применять также специализированные стандарты и руководства, появившиеся на свет благодаря широкомасштабным мероприятиям по защите критической инфраструктуры, проводимым в США, такие как «Urgent Action Standard

1200 – Cyber Security», Cryptographic Protection of SCADA Communications, Part 1: Background, Policies, and Test Plan» и многие др. [2].

## Заключение

Защита ПС критической инфраструктуры от кибератак постепенно становится одним из высших приоритетов в обеспечении государственной безопасности. Когда повышенный уровень террористической угрозы сочетается со стремительно возрастающим уровнем зависимости общества от ПС, этот вопрос стоит особенно остро и требует от правительства принятия скоординированных всеобъемлющих мер. И хотя кибертеррористических актов на сегодня фактически зафиксировано не было, угроза представляется вполне реальной.

Большинству ПС присущи уязвимости, характерные и для других ИТ систем, а также уязвимости, являющиеся для них специфическими. Для обеспечения адекватной защиты таких систем их разработчики должны будут предпринять значительные усилия по повышению уровня защищенности своих продуктов, встраиванию в них функций безопасности в соответствии с требованиями специализированных профилей защиты с последующей их сертификацией. Владельцы и организации, эксплуатирующие ПС, должны будут изменить свое отношение к вопросам информационной безопасности и существующую систему приоритетов в этой области.

Важнейшая роль в решении вопросов безопасности ПС критической инфраструктуры будет принадлежать правительству, органам государственной безопасности и антитеррористическим структурам. Для проведения соответствующих научно-исследовательских работ, разработки стандартов, методологий и средств защиты ПС потребуются значительные объемы государственного финансирования, масштабные государственные программы и правовое регулирование.

## Литература


1. Астахов А. Реалии и мифы кибертерроризма. // Открытые системы. 2003. № 5.
2. Andrew Hildick-Smith. Security for Critical Infrastructure SCADA Systems, GSEC Practical Assignment, Version 1.4c, Option 1, 2005.
3. Gellman, Barton, Cyber-Attacks by Al Qaeda Feared, Washington Post, 27 Jun 2002.
4. Digital Bond Inc., Control Center Protection Profile for Industrial Control Systems Version 0.50, Draft, 17 Feb 2004.
5. The National Strategy To Secure Cyberspace, Feb 2003.

# МЕТОД ОЦЕНКИ ДОСТУПНОСТИ ПРОГРАММНО-АППАРАТНЫХ КОМПЛЕКСОВ, ПОСТРОЕННЫХ С ПРИМЕНЕНИЕМ ТЕХНОЛОГИЙ ГОРЯЧЕГО РЕЗЕРВИРОВАНИЯ КОМПОНЕНТОВ

**В.В. Заря, А.А. Протченков, Е.В. Симаков**
**Научный руководитель – д.т.н., профессор Ю.А. Гатчин**

Рассмотрен метод оценки доступности информационных систем, состоящих из многих компонентов, использующих горячее резервирование. Приведено сравнение показателей доступности для систем без аппаратной избыточности и систем с дублирующимися элементами.

## Введение

В настоящее время к информационным системам, играющим критически важную роль в операционной деятельности организаций, предъявляются повышенные требования с точки зрения надежности. На этапе проектирования таких систем, как правило, существует несколько рабочих конфигураций, каждая из которых в определенном аспекте выгодно отличается от остальных. В настоящее время для оценки доступности информационных систем применяются сложные математические модели – интервальные статистические модели [1], теория нечетких множеств [2]. В данной работе представлен упрощенный метод оценки доступности информационных систем, основанный на сравнениях числовых значений доступности. Приводится сравнительные оценки расчета доступности для систем, не содержащих механизмы повышения доступности, а также систем, подразумевающих дублирование всех компонентов и их горячее резервирование. Основываясь на данном методе, возможно обоснованное принятие решения в пользу той или иной архитектуры, а также выявление наименее надежных компонентов системы и их замена или переконфигурирование.

## Архитектура программно-аппаратного комплекса

Рассмотрим типовую архитектуру программно-аппаратного комплекса с применением многоуровневого подхода (рис. 1). Распределим все компоненты системы по уровням в зависимости от выполняемой функции. Такой подход объединяет все взаимозаменяемые компоненты в один уровень и в дальнейшем поможет при анализе системы с точки зрения ее доступности.

- Уровень пограничных устройств. Пограничное устройство выполняет функции экранирования. Таким устройством может быть аппаратный межсетевой экран или маршрутизатор. Одним интерфейсом устройство подключено к внешней сети, другим – к исследуемому программно-аппаратному комплексу. На данном уровне происходит фильтрация трафика, приходящего в систему.
- Уровень коммутационных устройств. Обеспечивает физическое взаимодействие между компонентами системы, такими как серверы приложений и серверы баз данных.
- Уровень серверов приложений. Обрабатывает запросы пользователей к исследуемому программно-аппаратному комплексу. Инициирует запросы к серверам СУБД.
- Уровень серверов СУБД. Серверы СУБД управляют доступом к данным.
- Уровень систем хранения. Дисковый массив, на котором непосредственно хранятся данные.

Как видно из рис. 1, система не содержит избыточных компонентов и не подразумевает резервирования. При выходе из строя любого компонента приходит в неработоспособное состояние.

Для расчета доступности будем использовать принципы [3].

Множество компонентов, при выходе из строя любого из которых система теряет возможность предоставлять требуемый сервис, будем называть последовательными. Расчет доступности системы, состоящей из последовательных элементов, можно производить по формуле

$$A_S = \prod_{i=1}^{n} A_i \,, \qquad (1)$$

где $A_S$ – доступность всей системы, $A_i$ – доступность каждого последовательно соединенного компонента. Для системы, состоящей из двух компонентов, схематичное изображение расчета доступности показано на рис. 2.

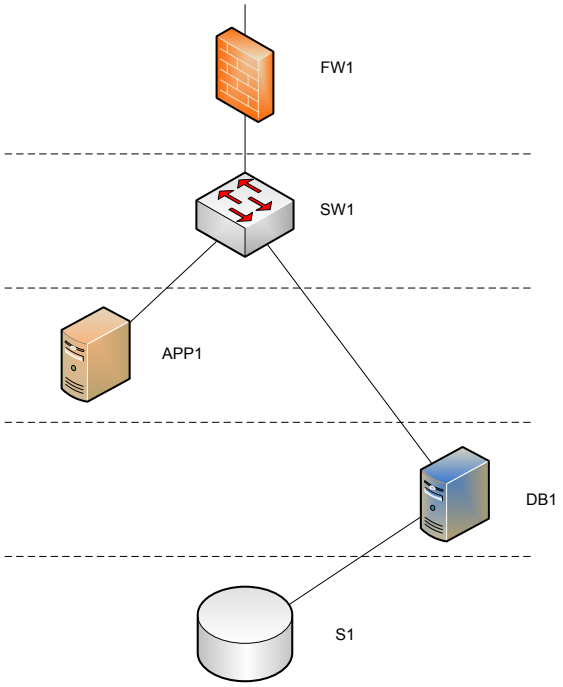



Рис. 1. Архитектура программно аппаратного комплекса без резервирования компонентов

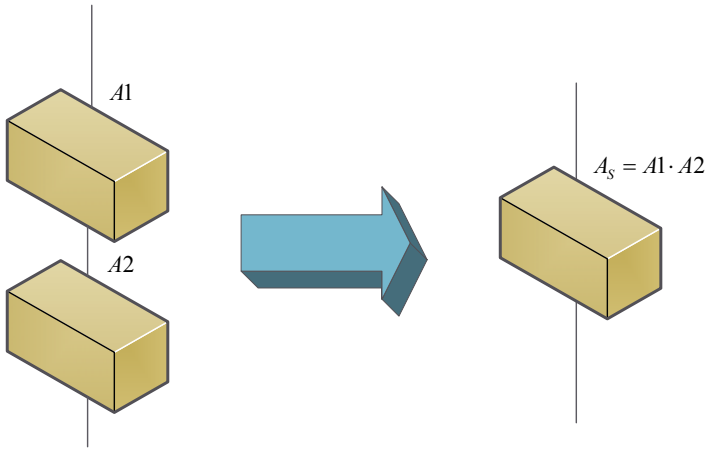



Рис 2. Вычисление доступности системы, состоящей из последовательных компонентов

Как видно из формулы (1), с увеличением числа компонентов системы экспоненциально уменьшается ее надежность (рис. 3, кривая Serial).

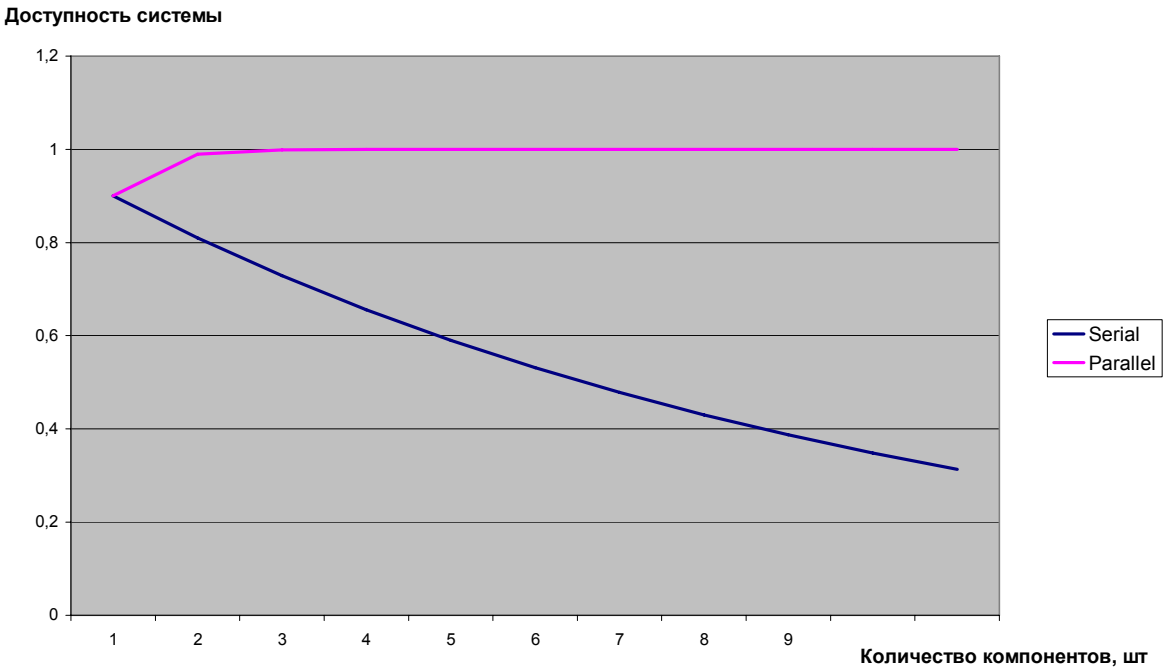



Рис. 3. Графики доступности системы с последовательными и параллельными компонентами

Множество компонентов, при выходе из строя одного из которых система не теряет возможность предоставлять требуемый сервис, будем называть параллельными. Расчет доступности системы, состоящей из параллельных элементов, можно производить по формуле

$$A_P = 1 - \left( \prod_{i=1}^{n} (1 - A_i) \right), \qquad (2)$$

где $A_P$ – доступность всей системы, $A_i$ – доступность каждого параллельно соединенного компонента. Схематично вычисление доступности системы изображено на рис. 4.

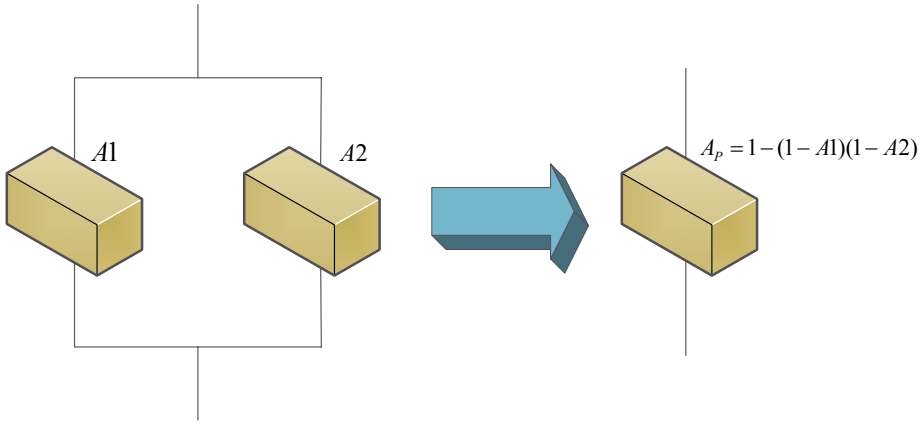



Рис. 4. Вычисление доступности системы, состоящей из параллельных компонентов

Зависимость доступности системы от количества параллельных компонентов изображена на рис. 3, кривая Parallel.

Таким образом, проанализировав существующую систему на предмет ее доступности, можем придти к выводу, что ее доступность зависит от значения доступности каждого компонента, а также от общего количества компонентов. Так как в существующей системе не предусмотрено резервирование, то сбой любого из компонентов вызовет перебои в пре-

доставлении сервиса. Существующая конфигурация не отвечает предъявляемым требованиям высокой доступности. Например, если значение доступности каждого компонента, кроме дискового хранилища, положим равным 0,99, а доступность дискового хранилища (как устройства с повышенной надежностью) – 0,999, то общая доступность системы равна $A_S = 0{,}99^4 \cdot 0{,}999 = 0{,}9596$, что недопустимо мало.

## Предлагаемая архитектура

Для уменьшения рисков простоя применяется система, использующая аппаратную избыточность и механизмы горячего резервирования компонентов. Схема предлагаемого решения приведена на рис. 5. Общая идеология такова, что при выходе из строя одного компонента происходит перераспределение нагрузки, так что его функции выполняет резервный. Взаимные влияния между уровнями могут быть значительными, а могут отсутствовать вовсе. Например, при выходе из строя коммутатора SW1 основным коммутирующим устройством становится SW2. Весь трафик от серверов приложений (APP1, APP2) и серверов баз данных (DB1, DB2) обязательно проходит через коммутатор (SW2). Однако выход из строя коммутатора приводит к еще одному изменению в общей схеме коммутации: межсетевой экран FW1 оказывается «отрезанным» от системы и, соответственно, не маршрутизирует трафик между внутренней и внешней сетями. Современные межсетевые экраны обладают функциями определения отсутствия канала связи и корректно обрабатывают переключение нагрузки с главного на резервный. Таким образом, в данном примере уровень межсетевых экранов не изолирован от уровня коммутирующих устройств, а некоторым образом зависим от него. Кроме того, существует и обратная зависимость: при выходе из строя одного из межсетевых экранов происходит перераспределение нагрузки на коммутаторах.

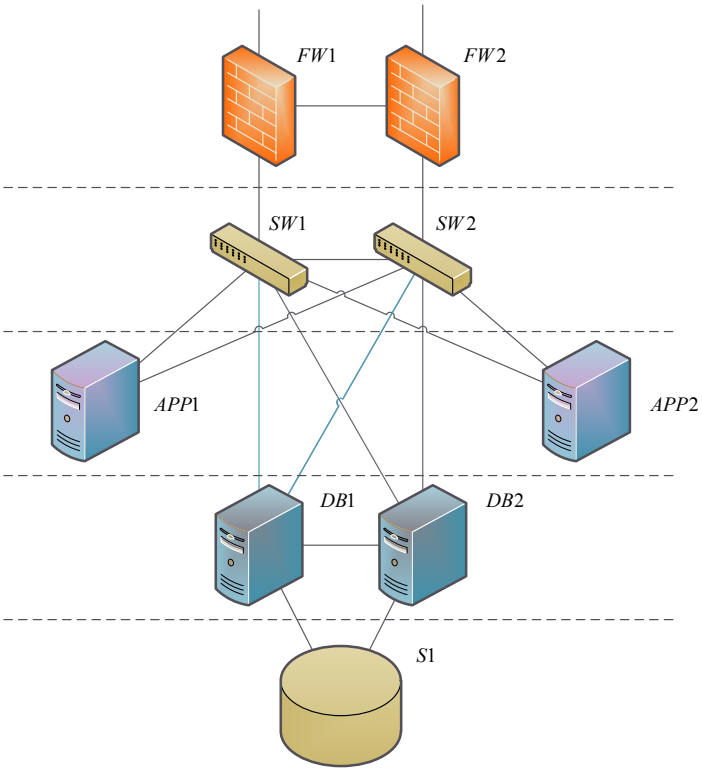



Рис. 5. Архитектура высокодоступного решения

Однако указанная закономерность не справедлива для всех компонентов системы. Например, выход из строя одного из коммутаторов никак не скажется на функциониро-

вании серверов приложений. Благодаря избыточным связям каждого сервера приложений с каждым коммутатором (каналы связи APP1-SW1, APP1-SW2, APP2-SW1, APP2-SW2) маршруты пакетов начнут проходить через резервный коммутатор совершенно прозрачно для серверов приложений. Такая конфигурация может быть реализована с помощью технологии объединения нескольких физических сетевых интерфейсов в один логический [4].

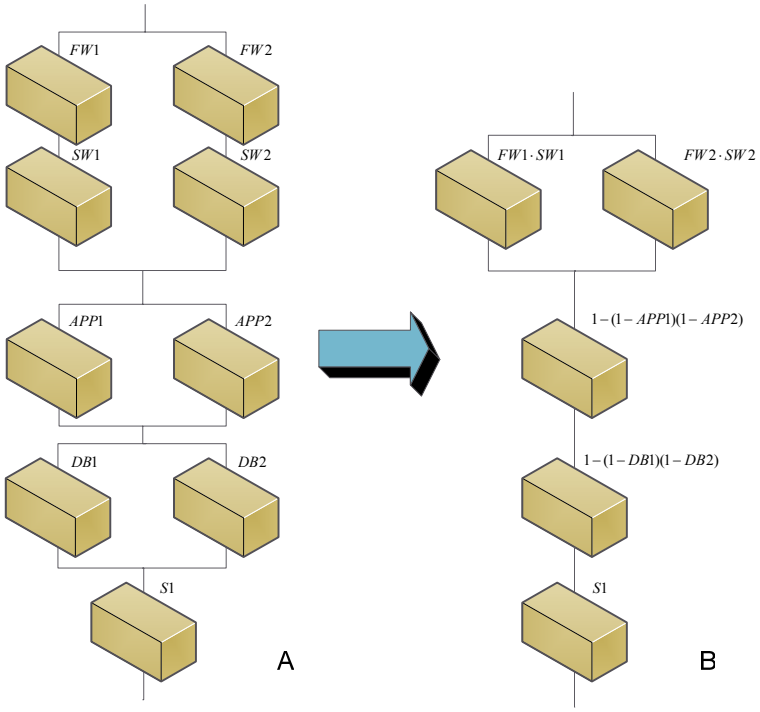



Рис. 6. Преобразование схемы компонентов

Рассмотрим более детально каждый уровень данной системы и его конфигурацию.

- Уровень пограничных устройств. Два межсетевых экрана объединены в высокодоступный кластер. Выход из строя одного вызывает перенаправление трафика на другой. МСЭ могут функционировать в режиме отказоустойчивости «активный/активный», в этом случае они выполняю еще и балансировку нагрузки, или в режиме «активный пассивный» – одно устройство активно, второе в состоянии горячей замены. Данную конфигурацию поддерживают многие производители.
- Уровень коммутационных устройства. Добавление в схему второго коммутатора исключило коммутатор SW1 как единую точку отказа. Теперь при выходе из строя активного коммутатора трафик перенаправляется на резервный. Это происходит прозрачно для серверов приложений и серверов СУБД.
- Уровень серверов приложений. Серверы приложений объединены в кластер с балансировкой нагрузки и разделяют один виртуальный адрес. Все запросы к серверам приложений направляются на активный сервер кластера, резервный начинает обрабатывать запросы в случае неработоспособности основного.
- Уровень серверов СУБД. Серверы СУБД объединены в высокодоступный кластер, управляющий доступом серверов приложений к дисковому хранилищу.
- Уровень систем хранения. Дисковый массив, на котором непосредственно хранятся данные, теперь используется как общий ресурс узлов кластера. Ввиду того, что дисковые хранилища как класс устройств обладают повышенными показателями надежности и отказоустойчивости (что сказывается на их стоимости) и содержат в себе резервируемые компоненты, такие как блоки питания, процессоры, интерфейсы управления, резервирование дисковых хранилищ не производится.

Для схематического изображения вычислительного комплекса для последующего анализа доступности необходимо понимать взаимосвязи различных компонентов системы – возможность функционирования одного устройства без другого и изменения, происходящие в системе при переключении нагрузки с основного компонента на резервный.

После ряда преобразований полученной схемы переходим от варианта А к варианту В (рис. 6). Допустим, что доступность одинаковых компонент одинакова, т.е $FW1 = FW2 = FW$, $SW1 = SW2 = SW$, $APP1 = APP2 = APP$, $DB1 = DB2 = DB$, тогда доступность исследуемой системы равна

$$A = \left(1 - \left(1 - FW \cdot SW\right)^2\right) \cdot \left(1 - \left(1 - APP\right)^2\right) \cdot \left(1 - \left(1 - DB\right)^2\right) \cdot S .$$

Положив значения, доступности каждого компонента системы равными 0,99, имеем:

$$A = \left(1 - \left(1 - 0{,}99 \cdot 0{,}99\right)^2\right) \cdot \left(1 - \left(1 - 0{,}99\right)^2\right) \cdot \left(1 - \left(1 - 0{,}99\right)^2\right) \cdot 0{,}999 \quad = 0{,}9984$$

по сравнению со значением 0,9596 для системы без применения резервируемых компонентов. Таким образом, при удвоении комплекта оборудования, необходимого для построения высокодоступной системы, а также при добавлении возможности горячего резервирования компонентов с помощью программного обеспечения (кластерное ПО, объединение физических адаптеров в логические и т.д.) мы добились значительного увеличения коэффициента доступности – с 0,9596 до 0,9984. Если перевести полученные коэффициенты в среднее время простоя, то этот показатель уменьшился с 15 суток в год до 14 часов в год. Если учесть, что исходные показатели доступности компонентов были заведомо занижены, то полученный результат можно считать вполне удовлетворительным.

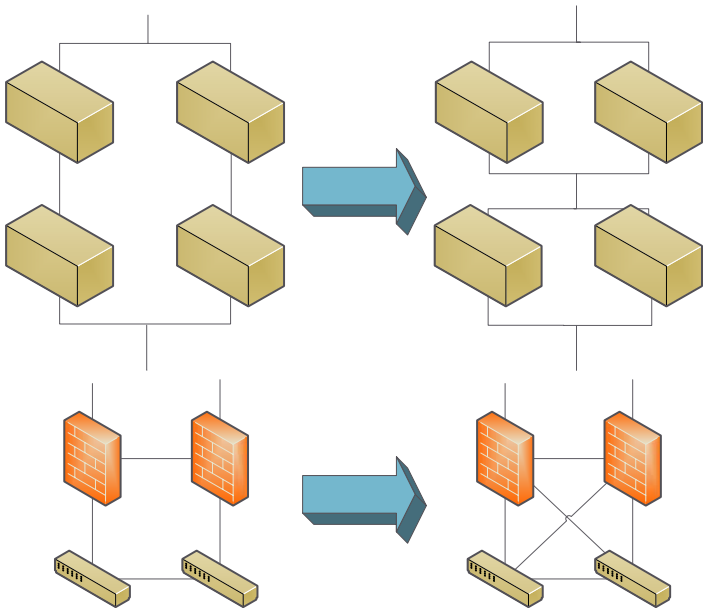


Рис 7. Преобразование схемы для повышения надежности

Проверим, есть ли возможность с минимальными затратами улучшить показатель доступности исследуемой системы. Исходя из рис. 5, А, можем заметить, что неоптимальным с точки зрения резервирования является участок FW1, FW2, SW1, SW2. В случае отказа одного из устройств первой ветки участка нагрузка целиком переносится на вторую ветку, оставляя работоспособное устройство первой ветки без нагрузки. Такой подход слишком расточителен с точки зрения резервирования ресурсов. Для устранения неисправности очевидно решение о преобразовании схемы (рис. 7).

В результате преобразований формула для вычисления доступности всей системы примет вид $A = \left(1-\left(1-FW\right)^2\right)\cdot\left(1-\left(1-SW\right)^2\right)\cdot\left(1-\left(1-APP\right)^2\right)\cdot\left(1-\left(1-DB\right)^2\right)\cdot S$, что при подстановке наших значений дает $A = 0{,}9986$ или 12 часов в год.

## Заключение

Метод оценки доступности программно-аппаратных комплексов может применяться при проектировании систем, одним из важнейших показателей которых является доступность. Описанный в работе прием оценки доступности системы был использован на практике при проектировании системы контроля и управления доступом на крупном промышленном предприятии.

Основным недостатком подхода является тот факт, что разработчику необходимо детально понимать принципы взаимодействия различных компонентов системы, а также обладать информацией о возможностях тех или иных технологиях и программных продуктах, непосредственно реализующих горячее резервирование. Необходимо точно ответить на вопрос о возможности такой конфигурации при использовании того или иного оборудования.

Описанная в работе система была построена с использованием оборудования таких производителей, как Cisco Systems, IBM, EMC. Для поддержки кластеризации применялись продукты Microsoft Cluster Services, EMC Power Path, Oracle Fail Safe, Cisco PIX OS.

## Литература


1. Кузнецов В.П. Интервальные статистические модели. М.: Радио и связь, 1991. 352 с.
2. Cai K.Y. Introduction to Fuzzy Reliability. Kluwer Academic Publishers, Boston, 1996.
3. Kok-Keong Lee. Building Resilient IP Networks. Indianapolis. Cisco Press. 2005. P.420–421.
4. Intel Advanced Networking Services With Ethernet Teaming. White paper. Texas. Dell Inc. 2005.

# РЕЗИСТИВНОСТЬ ВОДЯНЫХ ЗНАКОВ К JPEG ПРЕОБРАЗОВАНИЮ

**С.С. Кувшинов, О.В. Михайличенко, Н.Н. Прохожев**
**Научный руководитель – д.т.н., профессор А.Г. Коробейников**

В статье рассматриваются различные виды алгоритмов, применяющиеся при внедрении водяных знаков в изображение. Подробно описываются три алгоритма, являющихся характерными представителями своего вида, и проводится оценка их устойчивости к JPEG сжатию.

## Введение

Водяные знаки (watermark) – это своего рода подписи, применяемые к изображениям главным образом с целью подтверждения прав собственности или авторских прав. Эти знаки не могут быть визуально обнаружены и могут рассматриваться как секретное сообщение, встраиваемое в изображение.

В статье рассматриваются три наиболее распространенных типа алгоритмов внедрения водяных знаков в изображения.

Алгоритмы пространственной области (Spatial Domain Algorithm) используют для встраивания водяных знаков непосредственные изменения значений параметров яркости и цветности изображений. Это обуславливает низкие затраты вычислительной мощности для операции встраивания и извлечения водяного знака.

Алгоритмы области изменяемого разрешения(Multiresolution Domain Algorithm). используют дискретное вейвлет-преобразование (DWT).

Алгоритмы частотной области (Frequency Domain Algorithm) манипулируют некоторыми характеристиками изображения в частотном диапазоне. Этот вид алгоритмов включает в себя два подтипа, один из которых будет рассмотрен более подробно. Эти алгоритмы используют дискретное косинусное преобразование (DCT). Данный вид алгоритмов наиболее близок к алгоритмам сжатия JPEG. Исходное изображение делится на блоки 8×8 пикселей и подвергается косинусному преобразованию, в результате получаются 8×8 матрицы коэффициентов $C_b(j,k)$, где $b$ – номер блока, а $(j, k)$ – позиция коэффициента. Первый коэффициент $C_b(0,0)$, также называемый DC, содержит информацию о яркости блока, остальные называются AC-коэффициентами.

## Алгоритм Langelaar

Данный алгоритм относится к алгоритмам пространственной области [1, 2]. Водяной знак состоит из битовой строки. Алгоритм манипулирует яркостью пикселов в блоках 8×8 пикселей. Это дает *XY*/64 возможных областей. Используется случайный выбор области, но никакого качественного выбора области не осуществляется.

**Механизм встраивания**. Однократно создается двоичный псевдослучайный образ размера 8×8, а затем используется во всех встраиваниях: $pat(x,y) \in \{0,1\}$, где $0 \le x$, $y$<8. Чтобы встроить бит s водяного знака в блок $B=\{l(x+x_0, y+y_0)$, где $0 \le x$, y<8, этот блок вначале расщепляется на два подмножества $B_0$ и $B_1$ посредством образа:

$B_0=\{l(x+x_0, y+y_0)$, pat$(x,y)=0\}$,

$B_1=\{l(x+x_0, y+y_0)$, pat$(x,y)=1\}$.

Для обеих категорий рассчитывается средние значения яркости: $l_0$ и $l_1$. Разность между ними двумя будет означать бит подписи, где $\alpha > 0$ служит в качестве порогового значения:

$l_0 - l_1 > + \alpha$, если s=1,

$l_0 - l_1 < - \alpha$, если s=0.

Если это соотношение не проявляется, то уменьшаем или увеличиваем яркость пикселов в категории $B_1$ до тех пор, пока оно не обнаружится.

Чтобы сделать алгоритм более устойчивым к JPEG-сжатию, блок искажается посредством DCT-преобразования, квантующего коэффициенты с определенным фактором качества $Q$, после чего выполняется обратное DCT-преобразование. Результатом является слегка измененный блок. Снова рассчитываем средние значения яркости для тех же двух категорий: $l_0$ и $l_1$

Теперь приходим к двойному условию:

$$\left.\begin{aligned} l_0 - l_1 > +\alpha \\ \hat{l}_0 - \hat{l}_1 > +\alpha \end{aligned}\right\} \text{ если s=1,}$$

$$\left.\begin{aligned} l_0 - l_1 < -\alpha \\ \hat{l}_0 - \hat{l}_1 < -\alpha \end{aligned}\right\} \text{ если s=0.}$$

**Механизм извлечения.** Чтобы извлечь бит $s$ водяного знака из блока $B$, рассчитываются средние значения яркости, посредством образа $l_0''$ и $l_1''$. Разность между ними двумя дает бит извлеченной подписи:

$s_k'' = 0$, если $l_0'' - l_1'' < 0$,

$s_k'' = 1$, если $l_0'' - l_1'' > 0$.

Теперь мы можем использовать общие методы для сравнения реконструированного S'' и исходного S водяных знаков.

## Алгоритм Corvi

Данный алгоритм относится к алгоритмам области изменяемого разрешения [3, 4]. Для обнаружения водяного знака требуется исходное изображение.

Водяной знак представляет собой последовательность чисел с гауссовым распределением с единичной дисперсией в нуле. Итерации DWT выполняются до уровня $I$, где приближенное изображение имеет порядка тысячи коэффициентов. С этими коэффициентами и осуществляются манипуляции.

**Механизм встраивания**. Перед тем как встроить водяной знак, рассчитывается среднее $\hat{\upsilon}_l$ от всех коэффициентов $\upsilon_l(i)$. Чтобы встроить бит $s_i$ водяного знака, коэффициент $\upsilon_l(i)$ заменяется на

$$\upsilon_l'(i) = \hat{\upsilon}_l + (\upsilon_l(i) - \hat{\upsilon}_l)(1 + \alpha s_i),$$

где α представляет собой параметр, определяющий интенсивность водяного знака.

**Механизм извлечения.** Чтобы проверить, содержит ли изображение $I''$ водяной знак $S$, рассчитывается приближенное изображение собственного уровня $I$. Далее вычисляются среднее $\hat{\upsilon}_l''$ и дисперсия $\sigma(\hat{\upsilon}_l'')$, а сигнал $S''$ рассчитывается как

$$S'' = \frac{(\upsilon_l''(i) - \hat{\upsilon}_l'')\dfrac{\sigma(\upsilon_l)}{\sigma(\upsilon_l'')} - (\upsilon_l(i) - \hat{\upsilon}_l)}{\upsilon_l(i) - \hat{\upsilon}_l}.$$

Учет среднего и дисперсии приближенного изображения делает алгоритм устойчивым к преобразованиям, улучшающим контраст, а также к изменениям яркости. Полученный таким образом сигнал $S''$ уже может быть сопоставлен с водяным знаком $S$ на основе использования общих методов. Авторы алгоритма используют корреляцию.

### Алгоритм Cox

Алгоритм относится к алгоритмам частотной области [5]. Водяной знак представляет последовательность *n* вещественных чисел, где каждое значение $s_i$ выбрано независимо, согласно нормальному распределению с нулевым средним значением и единичной дисперсией. Для встраивания водяного знака используются *n* AC-коэффициентов с максимальным размахом.

**Механизм встраивания**. Вот три различных формулы для встраивания бита $s_i$ коэффициент *ci*:

$$c_i^{'} = c_i + \alpha s_i, \quad (1)$$

$$c_i^{'} = c_i(1 + \alpha s_i), \quad (2)$$

$$c_i^{'} = c_i(e^{\alpha s_i}). \quad (3)$$

Уравнение 1 не может быть адекватным, если значения $c_i$ изменяются в широком диапазоне. Если дисперсия $\upsilon_i = 10^6$, то добавление 100 может оказаться недостаточно для установления метки, однако если $\upsilon_i = 10$, то добавление 100 исказит это значение до неприемлемого. Уравнения 2 и 3 более устойчивы к подобной разнице в масштабе.

**Механизм извлечения**. Алгоритм извлечения следует алгоритму встраивания. Вычисляются DCT-коэффициенты для $I''$, и для *n* AC-коэффициентов с максимальным размахом *I*, рассчитывается разность с соответствующими коэффициентами $I''$. Чконструтированное $S''$ может теперь быть сравнено с S с помощью общих методов.

### Сравнение устойчивости рассмотренных алгоритмов к JPEG сжатию

Каждый из рассмотренных алгоритмов требует параметра *a* – «коэффициент внедрения». Значение этого параметра для каждого алгоритма свое. Однако мы нуждаемся в методе определения этого параметра для различных алгоритмов, который позволит провести сравнение между алгоритмами. Используем следующий метод выбора параметров: водяной знак внедряется с различными значения параметров, а затем считывается, и определяется соотношение сигнал /шум (PSNR) и корреляция между оригинальным водяным знаком и считанным.

Для графиков, представленных на рис. 1, параметр *a* был выбран исходя из результатов, дающих максимальное значение сигнал/шум и значение корреляции 0.95.

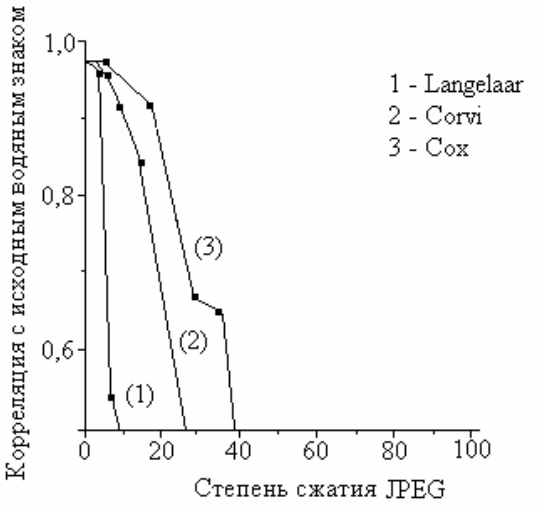



Рис. 1. Устойчивость водяных знаков к JPEG сжатию

## Заключение

Несмотря на то, что в алгоритме Langelaar с целью повышения резистивности к JPEG преобразованию осуществляется DCT преобразование, это явно не приносит желаемого результата, и устойчивость этого алгоритма оказывается очень низкой к такого рода воздействиям. Алгоритм Corvi оказался достаточно устойчивым, несмотря на то, что применяемый в нем метод вейвлетов значительно отличается от DCT-преобразования, используемого в JPEG. Следует ожидать, что данный алгоритм будет иметь наибольшую устойчивость к SPIHT сжатию, основанному на вейвлет-преобразовании. Алгоритм Cox, как и следовало ожидать, демонстрирует значительный уровень устойчивости, явно доминируя в этом аспекте среди других алгоритмов.

## Литература


1. Gerrit C. Langelaar, Reginald L. Lagendijk, and Jan Biemond. Realtime labeling methods for MPEG compressed video //. In 18th Symposium on Information Theory in the Benelux. Veldhoven, The Netherlands. 1997.
2. Gerrit C. Langelaar, Reginald L. Lagendijk, and Jan Biemond. Watermarking by DCT coefficient removal: A statistical approach to optimal parameter settings // In Ping Wah Wong and Edward J. Delp, editors, IS&T/SPIE Conference on Security and Watermarking of Multimedia Contents. California, USA. January, 1999. № 3657, pages
3. Ingemar J. Cox, Joe Kilian, Tom Leighton, and Talal Shamoon. A secure, robust watermark for multimedia // In Ross Anderson, editor, Information Hiding: First International Worshop, volume 1174 of Lecture Notes in Computer Science, pages 183 – 206. Cambridge, UK. May 1996. Springer Verlag.
4. Ingemar J. Cox, Joe Kilian, Tom Leighton, and Talal Shamoon. Secure spread spectrum watermarking for images, audio and video // In Proceedings of the IEEE International Conference on Image Processing, ICIP '96, pages 243 - 246, Lausanne, Switzerland, September 1996.
5. Marco Corvi and Gianluca Nicchiotti. Wavelet-based image watermarking for copyright protection. In Scandinavian Conference on Image Analysis, 1997.

# НАШИ АВТОРЫ

**Адушкин Иван Михайлович** – аспирант кафедры проектирования компьютерных систем

**Александров Григорий Сергеевич –** аспирант кафедры проектирования компьютерных систем

**Бабков Александр Сергеевич –** студент кафедры проектирования компьютерных систем

**Бандура Александр Сергеевич –** аспирант кафедры проектирования компьютерных систем

**Барышев Михаил Викторович –** аспирант кафедры проектирования компьютерных систем

**Белов Эдуард Владимирович –** аспирант кафедры проектирования компьютерных систем

**Беляков Андрей Викторович –** студент кафедры проектирования компьютерных систем

**Боголюбов Данила Александрович –** аспирант кафедры проектирования компьютерных систем

**Буданова Анна Юрьевна –** аспирант кафедры проектирования компьютерных систем

**Великоруссов Юрий Анатольевич –** студент кафедры проектирования компьютерных систем

**Власов Виталий Владимирович –** студент кафедры проектирования компьютерных систем

**Годырева Анастасия Валерьевна –** студент кафедры проектирования компьютерных систем

**Григорьева Наталья Сергеевна –** сотрудник ОАО «Российский институт радионавигации и времени»

**Гришенцев Алексей Юрьевич –** аспирант кафедры проектирования компьютерных систем

**Гуськов Александр Александрович** – аспирант кафедры проектирования компьютерных систем

**Дрюков Николай Юрьевич** – аспирант кафедры проектирования компьютерных систем

**Елисеев Олег Валерьевич** – аспирант кафедры проектирования компьютерных систем

**Ермаков Николай Владимирович** – аспирант кафедры проектирования компьютерных систем

**Заря Виталий Валерьевич** – аспирант кафедры проектирования компьютерных систем

**Иваненчук Анастасия Юрьевна** – аспирант кафедры проектирования компьютерных систем

**Иванов Виктор Георгиевич –** студент кафедры проектирования компьютерных систем

**Исаева Евгения Владимировна** – аспирант кафедры проектирования компьютерных систем

**Киселёв Владислав Борисович –** аспирант кафедры проектирования компьютерных систем

**Когай Наталья Викторовна** – ассистент кафедры проектирования компьютерных систем

**Козак Владимир Александрович –** студент кафедры проектирования компьютерных систем

**Колмогорцев Евгений Леонидович –** аспирант кафедры проектирования компьютерных систем

**Косенков Павел Александрович –** студент кафедры проектирования компьютерных систем

**Крылов Вадим Анатольевич –** аспирант кафедры проектирования компьютерных систем

**Крюков Василий Викторович –** аспирант кафедры проектирования компьютерных систем

**Кувшинов Станислав Сергеевич –** студент кафедры проектирования компьютерных систем

**Лекомцева Мария Васильевна** – аспирант кафедры проектирования компьютерных систем

**Липатов Алексей Леонидович** – аспирант кафедры проектирования компьютерных систем

**Лопатнёва Надежда Вадимовна –** студент кафедры проектирования компьютерных систем

**Малинин Алексей Анатольевич –** аспирант кафедры проектирования компьютерных систем

**Масленников Михаил Вячеславович** – аспирант кафедры проектирования компьютерных систем

**Михайличенко Ольга Викторовна** – ассистент кафедры проектирования компьютерных систем

**Москаленко Станислав Владимирович** – аспирант кафедры проектирования компьютерных систем

**Нечаев Виталий Александрович** – аспирант кафедры проектирования компьютерных систем

**Николаева Татьяна Сергеевна –** студент кафедры проектирования компьютерных систем

**Осломенко Денис Валерьевич** – аспирант кафедры проектирования компьютерных систем

**Пазухин Андрей Владимирович** – аспирант кафедры проектирования компьютерных систем

**Петрова Елена Николаевна** – аспирант кафедры проектирования компьютерных систем

**Пирожникова Ольга Игоревна –** студент кафедры проектирования компьютерных систем

**Протченков Алексей Александрович –** аспирант кафедры проектирования компьютерных систем

**Прохожев Николай Николаевич** – аспирант кафедры проектирования компьютерных систем

**Пудов Денис Владимирович –** аспирант кафедры проектирования компьютерных систем

**Сарычев Дмитрий Юрьевич** – аспирант кафедры проектирования компьютерных систем

**Семенов Вениамин Александрович** – аспирант кафедры проектирования компьютерных систем

**Семенова Мария Александровна** – аспирант кафедры проектирования компьютерных систем

**Симаков Евгений Валерьевич** – аспирант кафедры проектирования компьютерных систем

**Скобелин Александр Александрович** – аспирант кафедры проектирования компьютерных систем

**Соловьев Валерий Владимирович** – аспирант кафедры проектирования компьютерных систем

**Соловьев Денис Викторович** – аспирант кафедры проектирования компьютерных систем

**Строганов Кирилл Витальевич** – аспирант кафедры проектирования компьютерных систем

**Стройков Илья Игоревич –** студент кафедры проектирования компьютерных систем

**Терентьев Андрей Олегович –** научный сотрудник кафедры проектирования компьютерных систем

**Тулякова Мария Сергеевна** – студент кафедры проектирования компьютерных систем

**Туранцев Дмитрий Сергеевич** – аспирант кафедры проектирования компьютерных систем

**Фёдоров Алексей Анатольевич** – аспирант кафедры проектирования компьютерных систем

**Федотов Андрей Сергеевич** – аспирант кафедры проектирования компьютерных систем

**Федулова Галина Викторовна** – студент кафедры проектирования компьютерных систем

**Фролков Владимир Николаевич** – кандидат технических наук, ассистент кафедры проектирования компьютерных систем

**Шапин Антон Валерьевич** – студент кафедры проектирования компьютерных систем

**Шилкин Денис Андреевич** – студент кафедры проектирования компьютерных систем

**Юдин Дмитрий Геннадьевич** – студент кафедры проектирования компьютерных систем

# СОДЕРЖАНИЕ

**Научно-технический вестник СПбГУ ИТМО**. Выпуск 40. **НАУЧНАЯ ШКОЛА «ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ, ПРОЕКТИРОВАНИЕ, ТЕХНОЛОГИЯ ЭЛЕМЕНТОВ И УЗЛОВ КОМПЬЮТЕРНЫХ СИСТЕМ». Труды молодых ученых** / Главный редактор д.т.н., проф. В.Н. Васильев. – СПб: СПбГУ ИТМО, 2007. 290 с.

**НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК СПбГУ ИТМО**
**Выпуск 40**

**НАУЧНАЯ ШКОЛА «ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ, ПРОЕКТИРОВАНИЕ, ТЕХНОЛОГИЯ ЭЛЕМЕНТОВ И УЗЛОВ КОМПЬЮТЕРНЫХ СИСТЕМ».**

**Труды молодых ученых**

Главный редактор
доктор технических наук, профессор
В.Н. Васильев
Дизайн обложки В.А. Петров, А.А. Колокольников
Редакционно-издательский отдел СПбГУ ИТМО
Зав. РИО Н.Ф. Гусарова
Лицензия ИД № 00408 от 05.11.99.
Подписано в печать 10.10.07.
Заказ 1065. Тираж 100 экз.