

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
(наименование института полностью)

Кафедра «Прикладная математика и информатика»
(наименование кафедры)
09.04.03 Прикладная информатика
(код и наименование направления подготовки, специальности)

Информационные системы и технологии корпоративного управления
(направленность (профиль)/специализация)

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

на тему «Анализ методов и алгоритмов внедрения мобильной корпоративной
информационной системы на базе платформы 1С»

Студент О.Ю. Пушкирев
(И.О. Фамилия) _____ (личная подпись)

Научный
руководитель О.М. Гущина
(И.О. Фамилия) _____ (личная подпись)

Руководитель программы д.т.н., доцент, С.В. Мкртычев _____
(ученая степень, звание, И.О. Фамилия) (личная подпись)
«_____» 20____ г.

Допустить к защите
Заведующий кафедрой к.т.н., доцент, А.В. Очеповский _____
«_____» 20____ г. (личная подпись)

Тольятти 2019

СОДЕРЖАНИЕ

| | |
|--|----|
| ВВЕДЕНИЕ..... | 4 |
| 1 ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПОСТРОЕНИЯ И ВНЕДРЕНИЯ МОБИЛЬНОЙ КОРПОРАТИВНОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ..... | 8 |
| 1.1 Мобильные корпоративные системы и их роль в деятельности организации | 8 |
| 1.2 Анализ платформ реализации мобильной корпоративной информационной системы | 15 |
| 1.3 Внедрение информационной системы 1С «Консолидация холдинга» на предприятии..... | 19 |
| 1.3.1 Исходные данные по предприятию..... | 19 |
| 1.3.2 ИТ-архитектура в холдинге | 21 |
| 1.3.3 ИТ 1С на уровне холдинга | 22 |
| 1.4 Обобщение и оценка результатов..... | 23 |
| 1.5 Выбор методов и подходов аналитического и математического аппаратов исследования..... | 27 |
| 1.6 Архитектурные решения в области построения мобильной корпоративной информационной системы | 34 |
| 2 РАЗРАБОТКА АЛГОРИТМОВ ВНЕДРЕНИЯ МОБИЛЬНОЙ КОРПОРАТИВНОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ | 44 |
| 2.1 Проблемы корпоративной мобильности предприятия | 44 |
| 2.2 Методы и методики выбора внедряемых облачных сервисов в организации | 44 |
| 2.3 Модель и метод внедряемых мобильных корпоративных информационных систем..... | 50 |
| 2.4 Технологии тестирования и отладки программного продукта | 61 |
| 2.4.1 Разработка и тестирование интерфейса приложений | 61 |
| 2.4.2 Тестирование мобильных приложений | 62 |
| 2.4.3 Тестирование базы данных | 63 |
| 2.4.4 Генерирование данных для тестирования РБД..... | 65 |

| | |
|---|------------|
| 2.4.5 Прогнозирование развития РБД | 66 |
| 3 РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТАЛЬНОГО ИССЛЕДОВАНИЯ | 68 |
| 3.1 Методика выполнения операций управления мобильностью | 68 |
| 3.2 Математическая постановка задачи | 70 |
| 3.3 Разработка приближенного и точного алгоритмов решения задачи о покрытии на основе рангового подхода и проведение экспериментального исследования..... | 76 |
| 3.4 Подготовка и оформление доказательной базы диссертационного исследования согласно рабочей гипотезе..... | 82 |
| 3.5 Обработка результатов экспериментальных исследований в соответствии с рабочей гипотезой | 84 |
| ЗАКЛЮЧЕНИЕ | 89 |
| СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ | 94 |
| ПРИЛОЖЕНИЕ А Сопровождение и развитие систем 1С | 100 |
| ПРИЛОЖЕНИЕ Б Решение намеченных задач | 101 |
| ПРИЛОЖЕНИЕ В Листинг программы..... | 102 |

ВВЕДЕНИЕ

Появление мобильных устройств стало не только переворотом в развитии, но и кардинально изменило жизнь современных людей. Теперь нет необходимости находиться в офисе, или ездить в разные города для обсуждения важных корпоративных вопросов, поскольку использование современных мобильных устройств - главным образом смартфонов и планшетов - позволило людям получать удаленный доступ к своим данным и почты, общаться на расстоянии в реальном времени и сохранять информацию на виртуальных носителях. Однако расширение возможностей и широкий спектр использования мобильных устройств приводит к появлению все новых угроз. Учитывая, что рынок именно таких устройств является относительно молодым, то и программных средств для качественного администрирования и защиты от вирусов и утечки данных не так много. В основном проблемами защиты своих мобильных устройств занимаются производители – они пользуются базовыми средствами защиты, предусмотренные в современных операционных системах, таких как iOS, Android, Windows Phone. Но этих средств может быть недостаточно для качественного ведения корпоративной деятельности. Это привело к появлению специального ПО, обеспечивающего безопасность и конфиденциальность передаваемой информации, исключение утечек персональных и корпоративных данных.

Взаимодействие через мобильные системы всегда связано с определенным риском несанкционированного доступа. По данным исследований, в крупных компаниях в среднем происходит 91% краж мобильных устройств, 53% краж в офисе, 24% потеря мобильных устройств на корпоративных мероприятиях, 50% пользователей сохраняли персональные данные, номера банковских счетов на мобильных устройствах и другую конфиденциальную информацию.

Сотрудников часто озадачивают комплексы администрирования мобильных систем, подобные BYOD или MDM – они вынуждают использовать ключ-код или разрешают удаленно форматировать устройства. Сотрудники, отвечающие за безопасность ИТ, должны отдавать своим действиям четкий

отчет и уважать приватность личных данных персонала, несмотря на наличие к ним доступа через эти системы.

Создание объектов интеллектуальной собственности, списков партнеров и клиентов, а также коммерческих секретов требует от компаний значительных усилий, поэтому одним из основных приоритетов в их работе является защита этих мобильных данных с помощью простых в использовании и легких в управлении средств защиты на основе политик.

Если устройство, принадлежит сотруднику, используется в рабочих целях, то возникает ряд дополнительных вопросов, касающихся конфиденциальности пользователя, контроля за устройством, порядка использования устройства, политик безопасности и защиты данных, на которые необходимо найти ответ для обеспечения защищенности коммерческих данных. Из вышесказанного следует, что задачи проработки системной методики и интеграции автоматизированных систем контроля доступа к КИС имеют высокую актуальность.

Теоретические аспекты эффективности и оценки внедрения исследованы в работах Кобко Л.И. и Колчанова В.Д., Берсенева Н.П., Якимовой О.Ю., Дороговой А.В., Кадушина А., Шурыгиной И.Г., Михайловой Н., Рычкова А.И. и ряда других авторов. В работах зарубежных авторов эти вопросы рассматривают, в частности, Hussein Al-Bahadili, Tan C., Ghassan F. Issa, Gulieva I.F., Guliev Ya. I., Sun L., Ryumina E.A., Shakir M. Hussain, Marston S..

Частные вопросы интеграции мобильных корпоративных информационных систем изучались в публикациях Citrix, Vmware, MobileIron и зарекомендовавшей себя на российских предприятиях компании Крипто-Про, а также в трудах M. Pierer, L. Tang, D. Krebs, T. Smolarek, E. Klein и K. Rhee..

Анализируя существующий курс развития современных мобильных корпоративных информационных систем обнаружено противоречие между возможностями технической реализации средств защиты информации и предъявляемыми к ним требованиями конфиденциальности и безопасности доступных методов передачи информации в мобильных корпоративных информационных системах с различными требованиями защищенности.

Объект исследования: процесс выбора МКИС для внедрения на предприятии.

Предмет исследования: методы и модели выбора мобильной корпоративной информационной системы для внедрения на предприятии.

Работа выполнена с целью разработки метода и программного обеспечения внедрения мобильной корпоративной информационной системы организации, модернизирующих показатели эффективности при планировании ИТ-стратегии компании.

Для достижения обозначенной цели в нашей работе будут рассмотрено решение следующих задач:

- анализ управления мобильной корпоративной информационной системы МКИС на базе приложений 1С;
- создание методики принятия решений, обосновывающих выбор сервисов, внедряемых в организации;
- разработка модели и метода интеграции в компании программных продуктов, использующих мобильные корпоративные информационные системы;
- разработка приближенного и точного алгоритмов решения задачи о покрытии на основе рангового подхода для обеспечения защиты мобильной корпоративной информационной системы;
- разработка и тестирование программного обеспечения.

Особое внимание удалено исследованию и синтезу элементов внедрения мобильной корпоративной информационной системы основе разработки алгоритмов решения задачи о покрытии на основе рангового подхода.

В нашей работе применялись методы тестирования мобильного программного обеспечения, системного анализа, рангового метода решения задачи о покрытии, анализ вариантов с несколькими критериями, модель масштабируемости и теория применения графов. В основу теоретической и методологической частей работы были положены методы научного познания, а

также исследования российских и иностранных ученых в сфере контроля корпоративной мобильности.

Предлагаемая к ознакомлению работа содержит введение, три раздела, заключение и библиографический список, включающий 52 источника, в том числе 5 на иностранном языке, 31 рисунок, 13 таблиц и 3 приложения.

1 ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПОСТРОЕНИЯ И ВНЕДРЕНИЯ МОБИЛЬНОЙ КОРПОРАТИВНОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ

1.1 Мобильные корпоративные системы и их роль в деятельности организации

Основным предметом управленческой деятельности является информация, которая благодаря технологическим разработкам приобретает новые свойства:

- становится все более доступной для сбора и перевода в цифровой формат;
- занимает важное место в сфере продуктов и услуг;
- ее невозможно однозначно определить с позиций стоимости и цены;
- прослеживается сокращение срока ее жизни и потери актуальности;
- риск ее раскрытия (безопасность, конфиденциальность)
- составляет существенную долю расходов для большинства предприятий.

Современные технологии связи ускорили появление глобального рынка, телекоммуникации и интегрированные системы позволяют управлять процессами независимо от расстояния. Расположение бизнеса перестало быть определяющим фактором достижения успеха. Мировые сетевые коммуникации сделали глобальное присутствие предприятия на рынке, а также способствовали появлению новых сетевых организационных форм бизнеса.

Современная эпоха основывается на усиленном обмене информацией, ее преобразовании, а затем осмыслении и использовании. Именно поэтому стратегия развития системы управления предприятием (СУП) имеет в первую очередь направляться на развитие информационного потенциала предприятия через развитие внутренней информационной инфраструктуры с соблюдением требований выбранной политики стандартизации в сфере ИКТ.

Развитие внутренней информационной инфраструктуры должно направляться на комплексный охват имеющихся взаимосвязей между

участниками бизнес-среды, а также качественное улучшение сложившихся и внедрение новых коммуникаций. Цель развития информационной инфраструктуры заключается в создании консолидации усилий и снижению трансакционных издержек на текущие работы. Несистемное вложение средств в развитие информационной инфраструктуры или использование неподходящих элементов и технологий, даже высокопроизводительных, как правило, заканчивается потерей ресурсов, преждевременным моральным старением или системной несовместимостью в организации совместной деятельности участников выполняемых бизнес-процессов.

Основные задачи внутренней информационной инфраструктуры такие:

- предоставление телекоммуникационных сетевых услуг;
- обеспечения и управления вычислительной техникой высокого уровня;
- управления общими клиентскими базами данных;
- применение экспертных ИТ для выявления новых полезных возможностей;
- поддержка локальных корпоративных сетей (Intranet, Extranet).

Стандартизация элементов и технологий внутренней информационной инфраструктуры необходима, чтобы достичь синергетического эффекта синхронизации составляющих СУП и переносом их влияния на объекты управления. Предприятию требуется логика организации информационной инфраструктуры, которая закреплена в принципах применения информации и ИТ. Интеграция бизнес-процессов позволяет комплексно представлять функционирование предприятия во внешней среде и легко конфигурировать спектр выполняемых работ. Основой интеграции процессов с точки зрения технологии является стандартизация. В общем для достижения желаемого уровня коммерческой и технической стандартизации информационной инфраструктуры имеет четко отражать логику организации баз данных и программного обеспечения предприятий, закрепленную в политике развития, взаимоотношений и технических альтернатив.

В условиях широкой диверсификации очень часто новое возникает не в

результате усовершенствования используемых механизмов управления, а вследствие создания новых комбинаций из них. Это объясняется тем, что современные средства управления сильно диверсифицированы, поэтому их конвергенция может обеспечить синергический эффект. Поэтому стратегия управления развитием информационной инфраструктуры должна предусматривать следующие мероприятия:

- обеспечение взаимосвязей предприятия с научно-исследовательскими институтами отраслевого направления для выявления новых или скрытых знаний;
- постоянное отслеживание технологических прорывов в таких сферах, как создание продукции, производственные процессы, информационное обеспечение;
- исследование возможностей конвергенции ранее независимых продуктов или технологий, чтобы получить принципиально новые свойства, что будет способствовать синергичной повышению эффективности деятельности предприятия;
- изучение возможностей усовершенствования цепей поставок, производства, отношений с покупателями за счет новых сетевых ИКТ в их верификацией состояния безопасности и правовой поддержки;
- исследование возможностей аутсорсинга фаз разработки нового продукта или принципиально новых специализированных производственных линий.

Согласно типологии цивилизаций А. Тоффлера сегодня происходит информационно-компьютерная революция, которая порождает так называемое информационное общество. Меняются не только формы, но и содержание деятельности предприятий. Очевидно, что информационная составляющая мировой экономики существенно менее инерционная сравнению с традиционной материальной. А в результате динамичного роста ее доли повышается вероятность появления и развития кризисных явлений в экономике в целом. В современных условиях характерными чертами бизнес-деятельности

отечественных предприятий стали низкая предсказуемость ситуации и высокий темп изменений, а такая ситуация часто генерирует проблемы, которые необходимо оперативно решать. В связи с этим трансформировались цели деятельности предприятия. Если раньше оно ориентировалось на получение максимальной прибыли, то сегодня все более важным становится обеспечение стабильного развития, уменьшения зависимости от внешних ресурсов, минимизации негативного экологического воздействия.

Современное предприятие динамично развивается, его компоненты приобретают новые свойства. Основными фундаментальными направлениями дальнейших исследований эволюции предприятий остаются следующие проблемы:

1. Реагирование на процессы хаотичности эволюции. Первичной хаотичной основой формирования СУП является рынок в общем смысле с его развитой инфраструктурой. Диссипативные, диффузные, рассеивающие факторы рынка повышают значение коммуникаций для обеспечения когерентности поведения подсистем, которые взаимодействуют. В таких условиях ослабления связей элементов внутри СУП может привести к распаду предприятия. Углубленное изучение этих проблем стало толчком для развития теории саморегулируемых организаций.

2. Планирование процессов коэволюции. Поскольку предприятия развиваются по-разному, неизбежно возникает проблема совместного развития, то есть коэволюции. Например, одним из основных требований эффективного развития СУП является формирование единого темпомира. То есть синтез простых эволюционирующих субструктур в одну сложную структуру прежде всего требует согласования общего темпа их эволюции, хотя интенсивность процессов в различных фрагментах может отличаться. Заметим, что, согласно общим закономерностям самоорганизации, сложноорганизованным предприятиям нецелесообразно навязывать своим подразделениям ручное управление, а стимулировать собственные, естественные тенденции развития и научиться в режиме резонанса поддерживать согласования с рыночными

преобразованиями. Такой подход совпадает с принципами восточной философии, как - соблюдение естественности "ненасилие над природой вещей".

3. Проблемы глобализации. Считают перспективным объединение усилий различных социально-экономических организаций на глобальном рынке, что, как правило, на практике сопровождается огромными потерями, отклонениями и задержками. Такой путь к интеграции различных субъектов в целостный организм не является равномерным и односторонним. Эволюционные преобразования в сложных форм организаций проходят через ряд циклов "распада - интеграции", "разделения - объединение", "торможение - ускорение процессов" и другие. Сегодня еще недостаточно изучены закономерности существования, коэволюции, конвергенции неоднородных территориально распределенных элементов с сохранением их особенностей, темпов развития, уровня организации тому подобное.

4. Прогнозирование процессов эволюции. Развитие социально-экономических организаций определяется не только прошлым, историей, традициями системы, сколько будущей структурной эволюции информационно-ориентированного общества. Сегодня наблюдаем многочисленные попытки моделирования спектров организационного структурного развития и целей саморазвития предприятий. Спектры новых СУП предприятий в открытых нелинейных средах исследованы лишь в отдельных аспектах в рамках локальных программ.

5. Управление направлениям ускорения эволюции. Развитие экономических систем показал, что, наряду со стихийным, возможен и целенаправленное развитие открытых социально-экономических образований, постепенное и длительное становление и развитие информационно-ориентированного цивилизованного рынка не обязательно.

Благодаря прогрессу информационно-коммуникационных технологий организации уже не являются замкнутыми структурами, малоэффективными стали предприятия с жесткими иерархическими методами построения СУП. Это объясняется существенным сближением производителя с потребителем, резким

ростом взаимозависимости и взаимодействия между ними. Именно это заставляет предприятия обеспечивать проведение эффективных сделок, чтобы оперативно реагировать на изменения рынка.

Эти проблемы объясняются мировыми тенденциями научно-технического развития и глобализации, в результате чего формируется новый уровень развития экономических отношений, для которого характерно увеличение дистанции между отдельными предприятиями по экономическому развитию. Эти особенности динамики процессов развития обусловлены следующими факторами:

- ускорение темпов развития мировой экономики: продолжительность отраслевых циклов уменьшилась с 7-5 лет в середине XX в. до 1,5-2 лет в наше время;
- рост требований потребителей к качеству продукции в условиях насыщенного рынка;

соответственно не только с требованиями стандартов, но и к скрытым ожиданиям;

- усиление интеграции функционально взаимосвязанных структур организаций как по вертикали - через увеличение размеров экономических систем для концентрации производственно-экономического потенциала, так и одновременно по горизонтали - для формирования гибких сетевых структур в условиях глобализации (например, аутсорсинга)
- повышение неопределенности в условиях экспоненциального роста объемов информационной экономики;
- удешевление информационных продуктов на фоне роста стоимости материального продукта в условиях уменьшения его ресурсной базы.

В процессе деятельности СУП должно обеспечить каждого менеджера такими релевантными методами управленаческого анализа для принятия управленаческих решений, чтобы процедуры управленаческого анализа сопровождали действия сотрудников на каждой фазе процесса управления, каждом звене процесса создания цепочки стоимости. Если в к информационным

обществе действовал принцип "каждый должен знать/уметь столько, чтобы в сумме все знали/умели все", то в информационном обществе действует принцип: "каждый должен знать/уметь столько, чтобы быть независимым в отношениях с другими". К важным признакам влияния информационного общества на переосмысление, а затем модернизацию СУП, целесообразно отнести:

- доступность интернет-технологий для каждого сотрудника независимо от места его нахождения;
- расширение возможностей использования информационных и интеллектуальных технологий для профессионального использования в управлении и производстве;
- прогрессирования интеллектуализации товаров/услуг;
- создание глобального пространства знаний, развитие рынка знаний;
- формирование основ интеллектуального капитала;
- глобализация конкуренции;
- многоуровневый и взаимосвязанный распределение рисков и ресурсов;
- децентрализация систем управления (тяготение к коллективных решений);
- повышение оперативности принимаемых решений;
- формирование качественно новых организационных структур (сетевых, виртуальных, с самообучением, интеллектуальных, эволюционных и т.д.).

НТП и в будущем существенно влиять на развитие мировой экономики. По нашему мнению, целесообразно выделить новые виды перспективных инновационных технологий, наиболее повлияют на рост мировой экономики в XXI в. (табл. 1).

Таблица 1 - Перспективные направления развертывания инновационных технологий

| Название | Содержательное наполнение |
|--|---|
| Технологии мобильного Интернета | В ближайшее десятилетие мобильные устройства смогут удовлетворить потребности практически всех желающих. Сплошной охват населения мобильным Интернетом повысит уровень организованныя бизнес-процессов и откроет новые спектры возможностей для ведения бизнеса. |
| Технологии интернета вещей | Основными источниками данных в XXI в. будут сенсоры, которые в реальном времени позволяют собирать и передавать информацию о любых изменениях. Результатом станет качественное контроль внешней среды, уменьшится вероятность негативных внешних воздействий, увеличатся возможности автоматизации массовых взаимодействий (например, дорожного движения) за счет внедрения массовых логистических систем для минимизации затрат. |
| Роботизированные и интеллектуальные технологии | Массовое распространение роботизированных систем в промышленности, которые могут работать круглосуточно семь дней в неделю и существенно уменьшают вероятность ошибок. В долгосрочной перспективе это минимизировать расходы и увеличит эффективность производства. |
| Печать сложных технических и органических систем | 3D-принтеры, способные печатать физические объекты, в ближайшие десятилетия станут особенно популярными. Такие устройства смогут использовать не только предприятия, но и обычные потребители дома. Уже сегодня ученые исследуют способы воспроизведения живых организмов. |
| Нанотехнологии | Спектр применения нанотехнологий чрезвычайно широк - производство структурированных материалов с заданными характеристиками (например, устойчивости, проводимости и т.д.), печать технологических плат или даже борьба с бактериями. |
| Биотехнологии | Обеспечивают использование живых организмов и биологических процессов в производстве, что откроет совершенно новую сферу для развития технологий (например, для манипулирования генами, лечение болезней, разработка новых продуктов). |
| Новые технологии производства электроэнергии | С развитием экономики доступных источников энергии становится все больше (например, термоядерный синтез). Благодаря новым технологиям можно добывать топливо, ранее недоступное. Ведутся работы по разработке источников возобновляемого топлива, которое в будущем сможет заменить углеводороды. |

1.2 Анализ платформ реализации мобильной корпоративной информационной системы

Проведенный анализ существующих на рынке программных продуктов показал, что в настоящее время существует достаточно большое количество программных продуктов, среди которых можно выделить:

- «1С: Предприятие»;
- «Парус - Предприятие».

«1С: Предприятие» - это специализированная объектно-ориентированная система управления базами данных (СУБД), предназначенная для решения широкого спектра задач автоматизации учета и управления, стоящих перед современными предприятиями, которые динамично развиваются [6].

Среди преимуществ «1С: Предприятие» можно выделить следующие:

- большое количество объектов метаданных;
- поддержка различных СУБД;
- простота использования;
- наличие огромного числа специалистов;
- поддержка со стороны компании 1С;
- поддержка типовых конфигураций.

«Парус - Предприятие» - программный продукт, который используется для автоматизации финансово-хозяйственной деятельности, коммерческих и государственных организаций, предприятий и управления [9].

Преимущества «Парус - Предприятие»:

- простота освоения;
- широкие функциональные возможности;
- высокая надежность функционирования;
- типовые настройки на различные типы предприятий;
- сокращение затрат на автоматизацию;
- возможность анализа учетных данных и информации в базе данных.

Данные программные продукты обеспечивают решение самых разнообразных задач, стоящих перед предприятием, и являются схожими по своей функциональности. Сравнительная характеристика существующих программных продуктов по некоторым критериям по 5-ти бальной шкале приведена в табл. 2.

Таблица 2 - Сравнительная характеристика существующих программных продуктов

| Критерий сравнения | 1С: Предприятие | Парус - Предприятие |
|--|-----------------|---------------------|
| 1 | 2 | 3 |
| Простота использования | 3 | 3 |
| Поддержка со стороны фирмы-разработчика | 5 | 5 |
| Типичные настройки на различные типы | 4 | 4 |
| Широкие функциональные возможности | 5 | 4 |
| Наличие большого количества специалистов | 5 | 4 |
| Поддержка различных СУБД | 4 | 3 |
| Большое количество объектов метаданных | 4 | 4 |
| Сокращение расходов на автоматизацию | 5 | 5 |
| Высокая надежность функционирования | 5 | 4 |
| Легкость учета | 4 | 3 |
| Оперативность обновления форм документов | 5 | 4 |
| Формирования шаблонов отчетов | 4 | 4 |
| Ведение анализа деятельности | 4 | 4 |
| Всего | 57 | 51 |

Как видно из сравнительной характеристики существующих программных продуктов (табл. 2), что «1С: Предприятие» имеет большую функциональность чем «Парус - Предприятие.

Пользовательский интерфейс, «Парус-Предприятие» - достаточно удобен и понятен, что позволяет достаточно быстро его освоить.

Интерфейс программного продукта представлен на рис. 1.

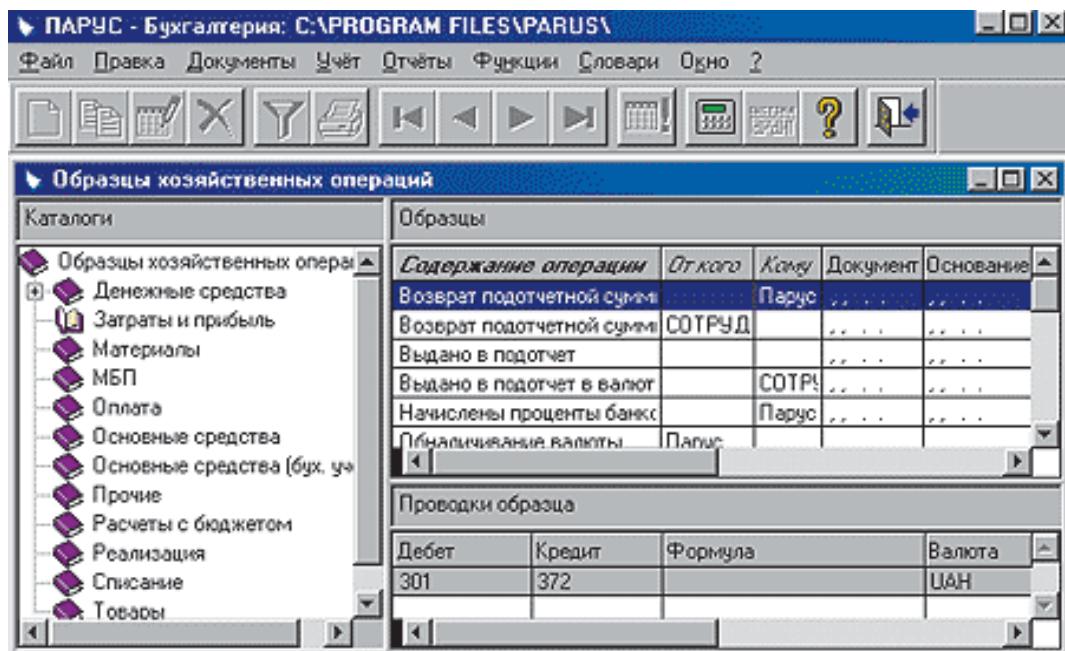


Рисунок 1 - Интерфейс программного продукта «Парус - Предприятие»

«1С: Предприятие» также предоставляет достаточно удобный пользовательский интерфейс, но его удобство и понятность превосходит многие другие программные продукты.

Интерфейс «1С: Предприятие» представлен на рис. 2.

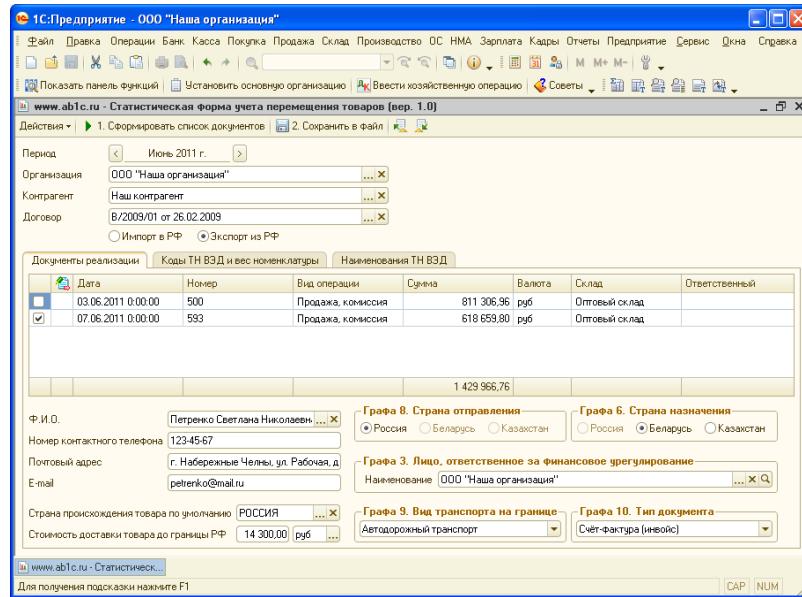


Рисунок 2 - Интерфейс программного продукта «1С: Предприятие»

Таким образом, автоматизация торгового предприятия с помощью таких программных продуктов как, «1С: Предприятие» и «Парус - Предприятие»

обеспечивают функциональность бизнес-процесса. Они способствуют увеличению эффективности управления, увеличения прибыли и объема продаж, дают возможность привлекать новых покупателей и увеличивать пропускную способность торговой точки, уровень обслуживания клиентов и тому подобное. Но стоит знать, что цена данных программных продуктов слишком велика для малых и начинающих предприятий.

1.3 Внедрение информационной системы 1С «Консолидация холдинга» на предприятии

1.3.1 Исходные данные по предприятию

ОАО «Татнефть» - сырьевое предприятие, обеспечивавшее в 1960–1970 годы треть всей нефтедобычи в СССР, стало одним из крупнейших в России вертикально интегрированных холдингов федерального и международного масштаба. Каждый год организация добывает порядка 26 млн. т нефти и 750 млн. куб. м газа, а также реализует программу стабилизации и увеличения объемов добычи углеводородного сырья, расширения масштабов деятельности и ресурсной базы. Реализуются проекты за пределами Республики Татарстан и Российской Федерации. [3, с. 12]

Основная часть получаемой продукции добывается с применением высокоэффективных технических средств. Компания стablyно утилизирует нефтяной газ на уровне, близком к 100%. В число перспективных направлений входит и выработка залежей сверх вязки (битумной) нефтяных продуктов, которые компания в режиме опытно-промышленных работ ведет на территории Татарстана. За все время деятельности холдинга, извлечено около 4 млрд. тон нефти, в дальнейшем кампания планирует достичь 5-миллиардного ракурса, за счет проведения геологоразведочных работ с целью оптимального извлечения сырьевого запаса из недр земли. В составе холдинга «Татнефть»:

- сеть объектов по реализации нефтяных продуктов;
- нефтехимические предприятия;
- нефтегазоперерабатывающие комплексы.

Сегодня компания представлена в 22 регионах Российской Федерации и имеет более шести сотен собственных заправочных комплексов. Компания активно вкладывает средства в техническое перевооружение и реконструкцию действующих производств. Реализуются и новые проекты, важнейшим из которых является строительство в Нижнекамске Комплекса нефтеперерабатывающих и нефтехимических заводов «ТАНЕКО».

Самый крупный инвестиционный проект состоялся в 2010 году в Нижнекамске, на заводе цельнометаллокордных шин, имеющих особую эффективность, долговечность и экологичность. Холдинг «Татнефть» первым вышел на мировую систему рыночных отношений, его акции были выставлены на международных биржах. Главным приоритетом работы акционерного общества является забота об экологическом состоянии мира. Системный комплекс экологического менеджмента и менеджмента профессионального здоровья и безопасности персонала компании имеют сертификацию по требованиям международного стандарта ISO 14001:2004 и спецификации OHSAS 18001:1999.

Высокий интерес в компании относятся к обеспечению безопасных и комфортных условий труда, дают возможности для профессионального развития – шагая по карьерной лестнице, социальных гарантитных условий для специалистов разного профиля. Холдинг «Татнефть» является ответственным налогоплательщиком, снабжает значительную долю доходов бюджета Татарстана, тем самым добавляет громадный вклад в развитие республики.

Для внедрения корпоративной системы в качестве объекта будет «Татнефть». В холдинге устойчиво формируются нефтедобыча, нефтеперерабатывающая, нефтехимическая, шинный комплекс, сеть АЗС и блок сервисных структур.

1.3.2 ИТ-архитектура в холдинге

Нефтедобывающий блок и структурные подразделения для учета документации, пользуются разными версиями SAP. Схема взаимодействия комитета по корпоративному управлению представлена на рис. 3

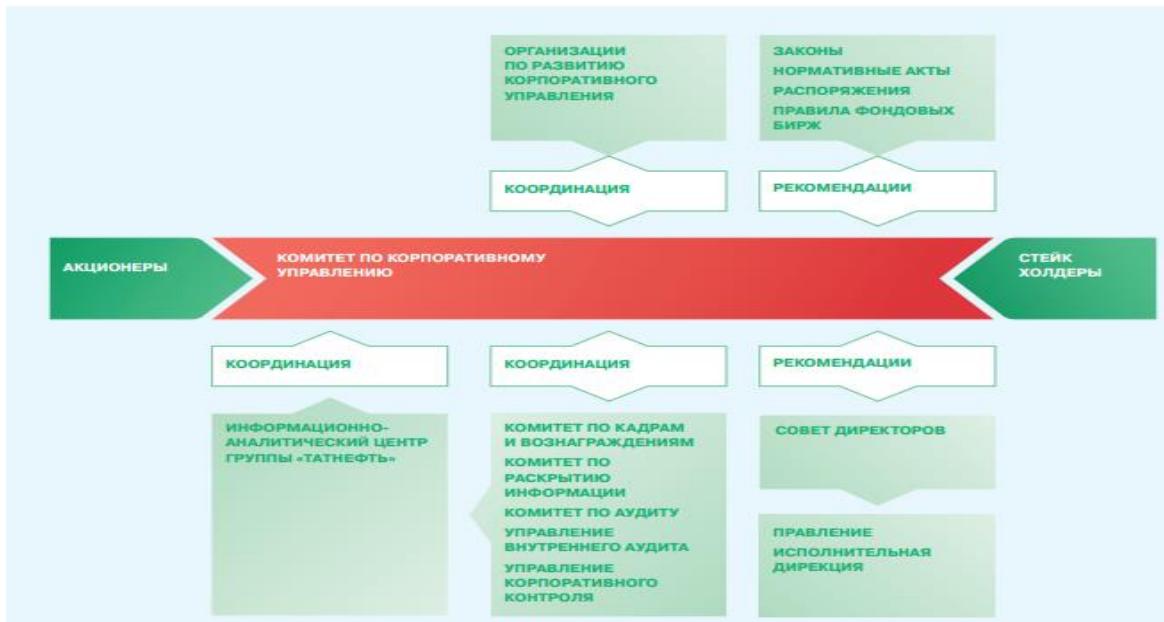


Рисунок 3 - Схема взаимодействия комитета по корпоративному управлению

Нефтегазодобывающее управление в каждой отрасли использует различные версии информационных систем, что значительно влияет на стоимость владения программами.

Разные платформы используют для различных операций в холдинге.

Дочерние организации – шинный комплекс, сервисный блок, предприятия соцкультбыта применяют платформу 1С. При этом эксплуатируют единые информационные системы для субхолдингов:

- Сеть АЗС;
- Сервисный блок (татнефтьМехСервис)
- Шинный комплекс.

Существенными недостатками действующей архитектуры на уровне ПАО «Татнефть»:

1. Более 200 информационных систем не связанных между собой;

2. Лоскутная автоматизация с использованием дорогостоящих систем;
 3. Дублирование систем, информационных потоков;
 4. Высокая стоимость владения и сложность применения;
 5. Невозможно остановить начатые процессы разработки, проблемное внедрение новых подсистем:
6. Отсутствие единой дорожной карты.

1.3.3 ИТ 1С на уровне холдинга

В 2017 году состояние дел по подготовке отчетности по международным стандартам выполнялась в затрудненном режиме:

- Выполнялась с привлечением специалистов PWC.
- Трудоемкий процесс работы с множеством таблиц MS Excel. На время подготовки ушло около 6 месяцев.
- Следствие работы проделана в сжатые сроки, имеет высокую вероятность ошибок.

Назревает необходимость внедрения информационной системы по подготовке отчетности всего холдинга. В холдинге система позволит охватить 2500 мест в 22-ух подразделениях.

Разные платформы 1С позволяют сократить время на формирование большинства документов. Будет приведен в порядок обмен юридических документов с 73 внешними контрагентами. Постоянно будет вести автоматическая проверка правильности введения данных и заполнения всех документов. [3, с. 10-12]

На повестке встает вопрос с оформлением отчетности холдинга по международным стандартам. Вариант обращения в научно-исследовательские центры SAP, который подберут оптимальный комплекс решений данной проблемы в холдинге.

Проанализированы преимущества 1С для системы построения корпоративной отчетности. Положено начало создания проекта по разработке системы подготовки отчетности по стандартам GAAP

Отчетность за 2016 год создана системы. Подтверждение аудиторов в достоверности отчетов, на подготовку ушло 3 недели. В настоящий момент в холдинге эксплуатируется блок информационной системы по подготовке отчетности.

1.4 Обобщение и оценка результатов

Новая подсистема редакции 1.5 позволила автоматизировать жизненный цикл инвестиционного проекта от инвестиционной идеи до завершения проекта. "1С:Управление холдингом 8" определила декомпозицию этапов проектов или программы взаимосвязанных проектов, указаны ответственные лица, сроки и используемые ресурсы.

В качестве ресурсов инвестиционных проектов могут выступать любые справочники конфигурации (например, "Сотрудники", "Номенклатура", "Внеоборотные активы"). Для ресурсов могут быть указаны натуральные и стоимостные параметры.

Иерархическая структура этапов может быть импортирована из Microsoft Project. Проект является глобальной аналитикой для управлеченческого учета. Благодаря возможности связать показатели бюджетов, отражение факта и резервирование по бюджетам движения денежных средств, доходов и расходов, движения ресурсов с этапами проектов система позволяет обоснованно и предельно точно вести бюджетирование проектов.

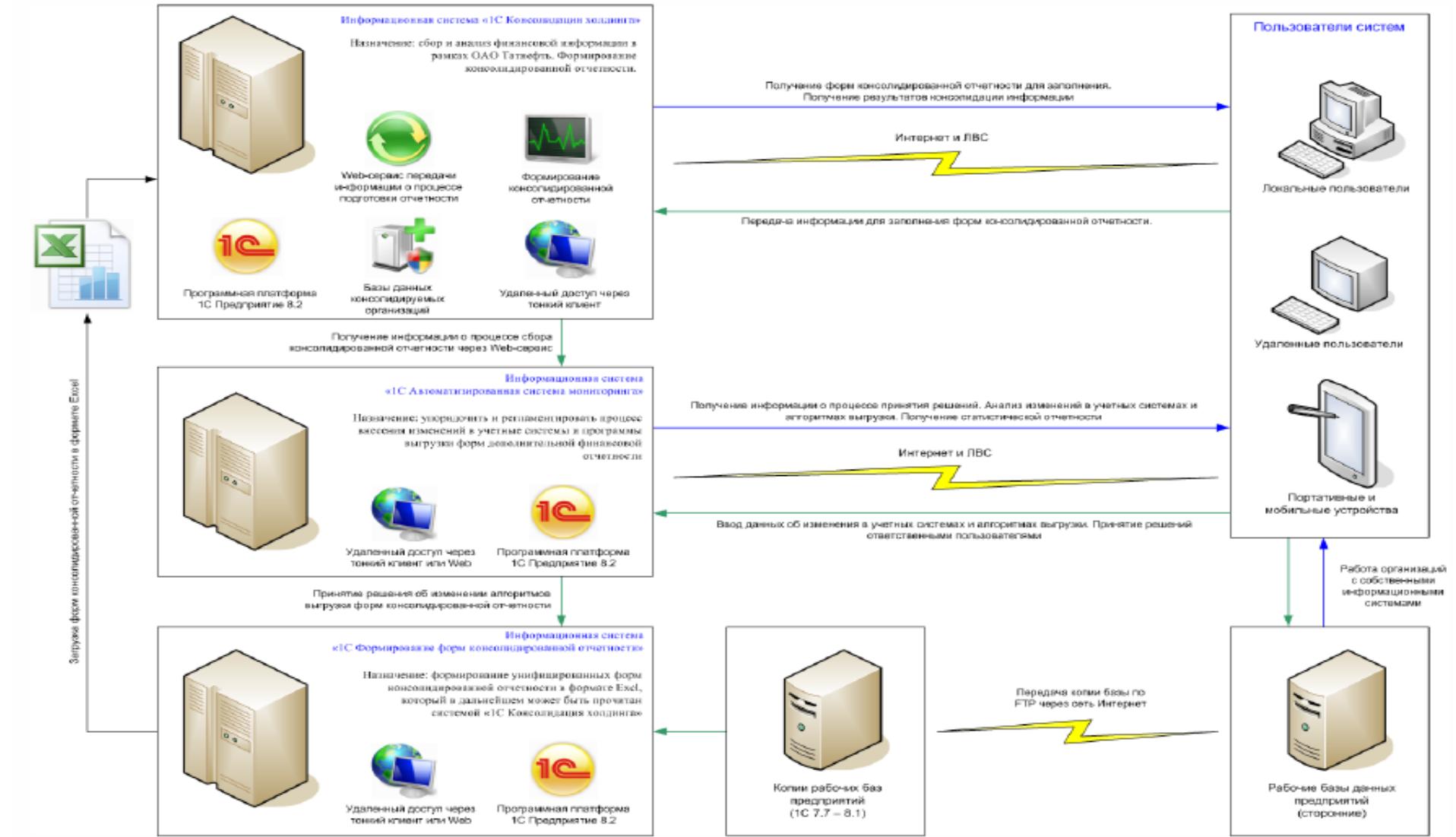


Рисунок 4 - Схема действия информационной системы

По данным бюджетов проектов автоматически рассчитываются их ключевые показатели эффективности (NPV, IRR, срок окупаемости, индекс прибыльности и др.).

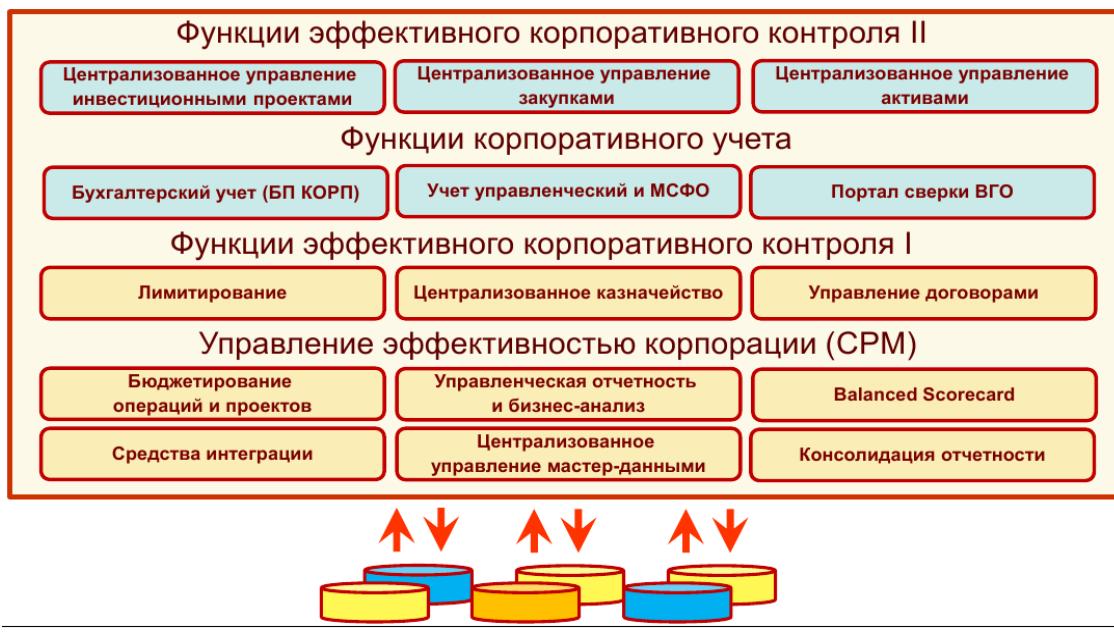


Рисунок 5 - Система учета и управление ДЗО

В целях автоматизации предыдущего состояния инвестиционной фазы прикладное решение позволяет определить критерии сравнения альтернативных проектов, провести их сравнительную экспертную оценку, включить в инвестиционную программу действительно оптимальный портфель проектов. [10, с. 22]

Удобные средства визуализации (диаграмма Ганта и диаграмма последовательности) позволяют наглядно представить сводную информацию о проекте. Непосредственно в диаграмме Ганта можно провести цепочку актуализации проекта, см. рис. 6:

- уточнить сроки этапов,
- разнести документы "Резервирование бюджета" по этапам,
- пересчитать бюджеты проекта по документам резервирования бюджетов,
- рассчитать ключевые показатели эффективности проекта.

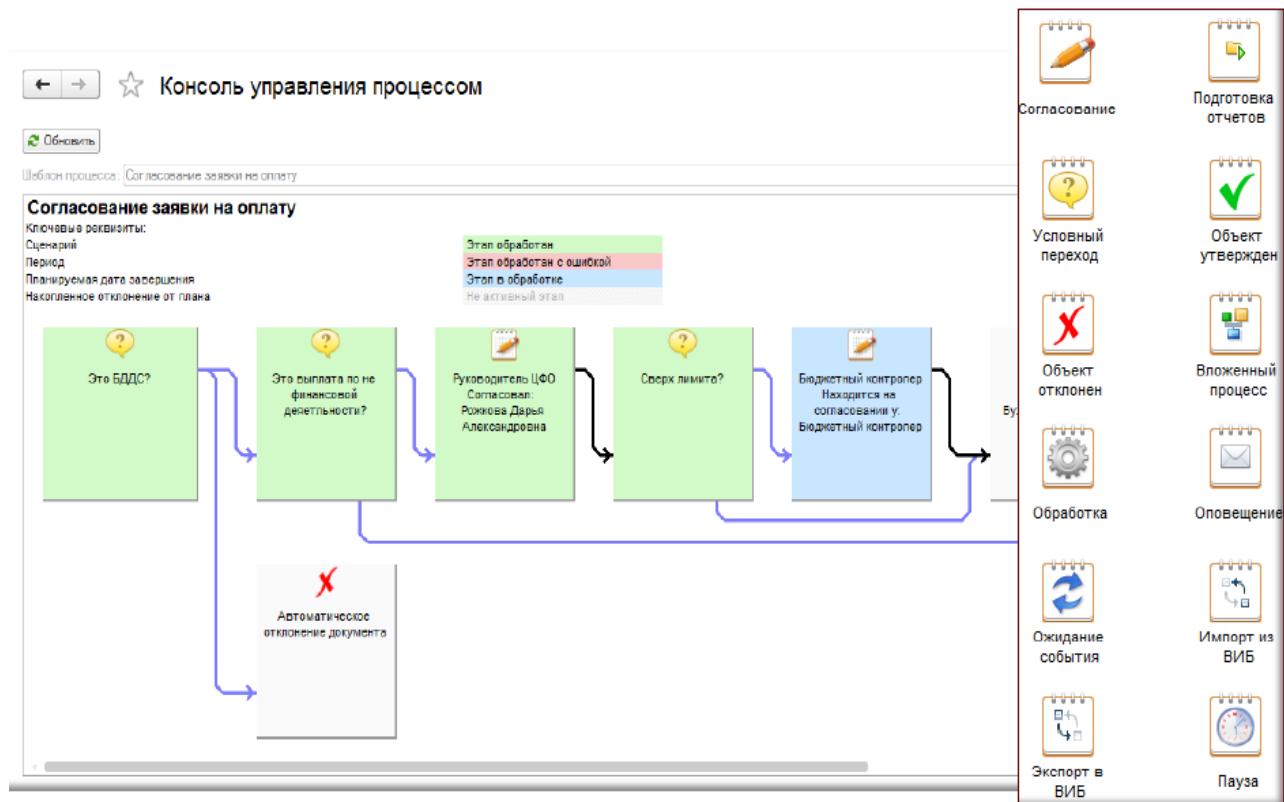


Рисунок 6 - Универсальные процессы 1С

Соблюдение концепции «Stage-Gate» в поэтапном создании портфеля проектов свести к минимуму риски крупнейших проектов холдинга: жизненный цикл проектов будет строиться в определенной последовательности, что позволит объективно рассматривать и решать различные вопросы на фазе их исполнения. Подсистема позволяет с разных позиций изучать проект, так как храниться информация о его состоянии, определена стадия эффективности проекта. При этом есть возможность сравнивать характеристику инвестиционных проектов всего холдинга.

В процессе были выделены достоинства системы:

- 1) удобства работы в программе:
 - три уровня аналитики плана счетов IFRS;
 - хранение макетных документов во внешних файлах;
 - настройка проведения форм по плану счетов;
 - доступный интерфейс;
 - настройка алгоритмов для расчета поправок;

- возможности вноса любых корректировок;
 - разделение уровней доступа к данным.
- 2) Гибкая система отчетов при подготовке данных:
- встроенные отчеты по проводкам с удобным отбором;
 - представление данных в текстовом и графическом виде;
 - отчеты для руководителей;
 - использование расшифровок.
- 3) Формирование финансовой отчетности:
- выгрузка данных в файл-шаблон в формате MS Word;
 - настройка выходных форм пользователем;
 - нет ограничений на получение и хранение отчетов;
 - проверка отчетности на ошибки и нестыковку данных.

1.5 Выбор методов и подходов аналитического и математического аппаратов исследования

Согласно Д.В. Карпову, заключение об интеграции мобильной корпоративной информационной системы принимается на основании следующих факторов:

1. Расширение предприятия привело к кардинальным изменениям, что в свою очередь напрямую повлияло на уровень эффективности текущей системы в худшую сторону;
2. В организации произошла смена направления деятельности и система уже более не может обработать появившиеся параметры;
3. Рейнжиниринг имеющихся практик компании предъявляет новые требования к функционалу и концепции текущей системы;
4. Используемая система потеряла актуальность, а модернизация невозможна ввиду слабой коммерческой выгоды для организации.

Помимо представленных факторов, каждая организация может иметь свои, уникальные причины для интеграции более актуальной КИС.

Когда первоначальный этап принятия решения об интеграции новой системы

утвержден, необходимо перейти к управлению интеграцией – для этого привлекаются профессионалы и специалисты, которые приступают к разработке корпоративного решения, стараясь найти удовлетворяющий заказчика баланс между качеством, временем разработки и стоимостью проекта [2-10]. Установленный проект интеграции необходимо согласовать с высшим руководством компании.

Авдошин С.М. выделяет следующие наиболее уникальные факторы управления проектом интеграции:

- необходимо оценить предполагаемую стоимость;
- необходимо оценить сложность процедур и методов интеграции;
- необходимо подвергнуть оценке предполагаемую экономическую эффективность;
- оценке также подвергаются риски и факторы неопределенности, как на этапе выбора, так и на этапе интеграции;
- производится анализ необходимых организационных и финансовых ресурсов.

На рисунке 7 представлен анализ управления интеграционного проекта с точки зрения системного процесса.

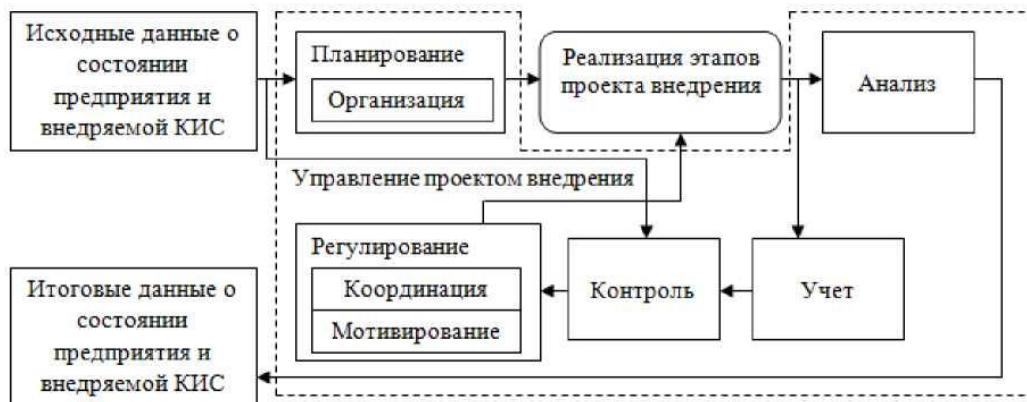


Рисунок 7 – Управление интеграционным проектом КИС

В ходе контроля проекта интеграции крайне важно учитывать вид проекта, его плановую экономическую эффективность и методы реализации. В данной работе мы уделим особое внимание этапу разработки стратегии интеграции

корпоративной информационной системы.

Высшему руководству организации необходимо решить, с какой именно стратегией интеграции работать. Среди таких проектов определяют:

- интеграция основного ядра внедряемой модели и разработка приложений-клиентов силами компании, корпоративная информационная система адаптируется лишь частично;
- интеграция готовой модели, полная адаптация корпоративной информационной системы в системы организации;
- полностью уникальное решение и самостоятельная разработка КИС, которая учитывает все потребности компании.

Чаще всего на современных предприятиях внедряются проекты первого и второго типов, в связи с большим выбором возможных вариантов российских и западных разработок корпоративных информационных систем [4]. Среди отечественных КИС выделяются «1С Предприятие», «Галактика», «Парус» и др.. При выборе определенной системы необходимо учитывать классификации, представленные на рисунке 8.



Рисунок 8 - Классификация корпоративных ИС

Классификация корпоративных ИС позволяет оценить решения, представленные на рынке и определиться с необходимыми функциями собственной

системы при самостоятельной разработке. Необходимо также учитывать особенности компании – при создании проекта важно совмещать их во время определения критериев и оценки альтернатив. Руководству компании неизбежно придется брать во внимание последствия и риски интеграции корпоративной информационной системы. Следовательно, возникает необходимость разработки методики выбора наиболее экономически эффективной КИС.

Ситуационный анализ появился достаточно давно, и на текущий момент рассматриваются два вида его реализации – широкий и узкий. Под первым видом понимают представление различных методов в основе ситуационного управления, под вторым – такой анализ используется для аккумуляции данных и дальнейшей их обработки в рамках имеющейся ситуации [5].

Решетников М.М. определяет ситуационный анализ как «изменение направления деятельности, задачей которого является преодоление препятствий, в рамках единой стратегии» [6]. Такой тип анализа используется для аккумуляции данных, их обмена и изменения, а также для подбора наиболее оптимального решения. Описанные методы используются для разработки стратегии интеграции корпоративной информационной системы, для поиска альтернативных решений, оценки и прогнозирования влияния интеграции на организацию и возможности масштабирования [7].

Процесс разработки стратегии интеграции корпоративной информационной системы с точки зрения описанного анализа представлен на рисунке 9.

Проект интеграции разрабатывается несколькими субъектами – приглашенными экспертами [10] – что обеспечивает создание мультиагентной системы взаимодействия. С этой позиции наиболее важно выделить следующие причины использования именно такого подхода:

- значительно сокращается время принятия решений в организации;
- эксперты могут дать независимую оценку имеющихся на предприятии проблем, что позволяет получить свежий и объективный взгляд на ситуацию;
- невозможно спрогнозировать непредвиденные и чрезвычайные обстоятельства, как и нельзя полностью исключать вероятность таких событий и

ситуаций;

- в организации могут возникать ситуации, независимые от проекта интеграции, влияющие на его реализацию;
- финансовые и кадровые ресурсы организации могут сыграть важную роль в реализации проекта;
- потребности каждой конкретной организации могут оказывать существенное влияние на выбор альтернативных решений и подбор требований, представленных к корпоративной информационной системе.

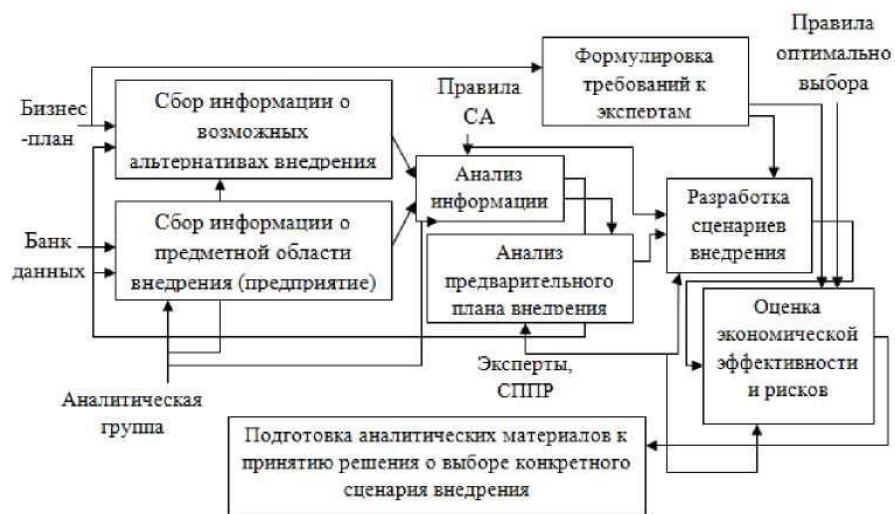


Рисунок 9 – Интеграция корпоративной информационной системы с точки зрения ситуационного системного анализа

На рисунке 10 представлена мультиагентная структура интеграции в рамках ситуационного анализа. Самостоятельно, каждый отдельный эксперт не сможет составить и интегрировать альтернативную корпоративную информационную систему в организации – это комплексная и сложная задача, которая предполагает различные стадии проектирования и взаимосвязь между внутренними и внешними агентами компании, выраженную в виде мультиагентной структуры взаимодействия.

Интеграцию корпоративной информационной системы на предприятии следует в первую очередь рассматривать, как инвестиции в будущее компании, поэтому крайне важно оценить потенциальные преимущества от улучшения конкурентноспособности организации на рынке, с учетом возможных рисков и

эффективности [8].

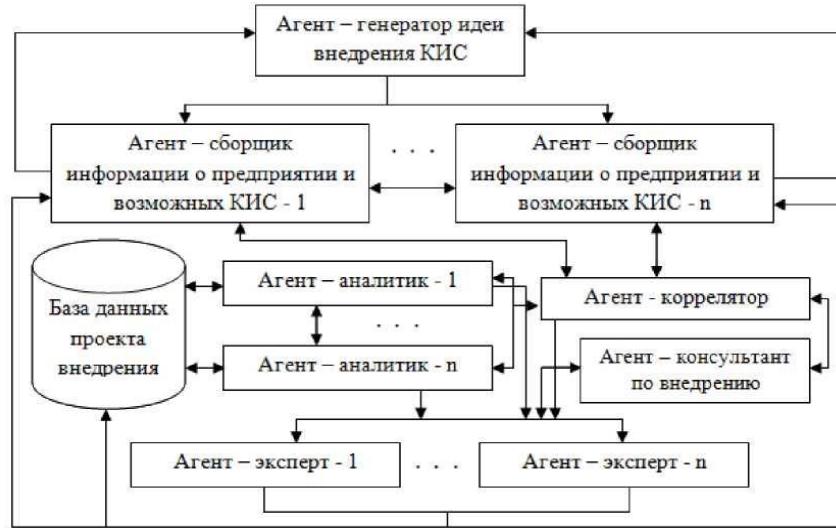


Рисунок 10 - Мультиагентная структура интеграции в рамках
ситуационного анализа

Результаты интеграции корпоративной информационной системы оцениваются по модели анализа эффективности на стадии разработки стратегии. Данная модель представлена на рисунке 11.



Рисунок 11 – Риски и эффективность интеграции КИС

Что же касается количественных методов, по которым можно оценить

эффективность внедрения КИС, то в большинстве случаев они рассматриваются с точки зрения вероятностной теории, как основной теории математического аппарата для просчета неопределенных событий. Можно выделить две группы методов, которые применяются для количественного анализа:

- вероятностные – в основе этого подхода лежит теория вероятности и ее аппарат, теория статистических решений и имитационное моделирование;
- невероятностные, они же детерминированные – изменяют показатели проекта в соответствии с возможными изменениями в ходе его реализации.

Подробнее остановимся на применяемых сегодня методах:

Вероятностные методы – ожидаемую эффективность можно корректно оценить, только если дать оценку результатам внедряемых решений. Анализ по данному методу предполагает определение вероятных пределов обозначаемых критических переменных и контрольные расчеты, которые допускают произвольные изменения переменных в границах их возможных значений. Этот метод позволяет рассчитать, какую стоимость следует ожидать от реализации проекта [10].

Метод сценариев основан на анализе отклонений результирующего показателя и совмещении его чувствительности.

Для попарного анализа альтернативных решений используется *метод анализа иерархий*, в котором берутся во внимание предполагаемые результаты решения задач, поставленных при разработке интеграции [9]. Согласно Саати, данный метод следует использовать, применяя шкалу оценок, где отражены преимущества и недостатки альтернативных систем в виде 5 базовых и 4 второстепенных суждений.

В случае, если при анализе эффективности возникает необходимость рассмотрения негативных факторов внедрения аналогичных решений, применяется *метод аналогов*. В рамках этого метода реализуется база данных, наполняемая информацией о процессах интеграции схожих проектов и мнениях непосредственных участников интеграции, скрытых трудностей, с которыми пришлось столкнуться при реализации. После сбора данных, они анализируются на

предмет взаимосвязей с ранее реализованными интеграциями, проводится структурирование возможных рисков процесса реализации.

Если текущие решения находятся в сильной зависимости от решений, с которыми пришлось столкнуться ранее, используют *метод создания «дерева решений»*. В рамках этого метода рассматриваются возможные варианты развития, ответвления событий с учетом предыдущих шагов реализации проекта интеграции корпоративной информационной системы.

Когда для корректной количественной оценки не хватает информации, прибегают к *методу экспертного анализа*. В данном случае при реализации КИС обращаются к опыту ИТ-персонала компании, который может негативно повлиять на принятие решений в связи с очень высоким риском субъективного оценочного суждения.

Ранее мы уже говорили, что при разработке стратегии интеграции КИС большую роль играет процент неопределенности, выраженный в следующих показателях [8]:

- последствия и результаты внедрения;
- состояние внешней среды;
- недостаточная четкость и обоснованность требований к эффективности;
- некорректные или неполные исходные данные;
- недостатки конкретной выбранной модели и варианта интеграции.

1.6 Архитектурные решения в области построения мобильной корпоративной информационной системы

Разработку информационной системы чаще всего определяет выбранная модель – на основании нее выбираются цели и задачи, реализуемые проектом. В качестве основных моделей итеративной разработки выделяют:

- Agile (Feature Driven Development, Scrum, Crystall);
- RUP – Rational Unified Process;
- PSP, TSP – Personal Software Process, Team Software Process;

- Microsoft Solutions Framework.

У каждой из представленных выше моделей есть особенности, и выделить наиболее универсальную среди них нельзя. Каждый отдельный случай требует особенного подхода, выбрать подходящую модель можно, руководствуясь следующими факторами:

- *Планируемое время использования* информационной системы – если в планы организации не входит долговременная поддержка ИС, то нет никакого смысла поддерживать подробную документацию. Это позволит сократить финансовые и кадровые издержки;
- *Отказоустойчивость* информационной системы – данный фактор стоит брать под особое внимание, если от работы ИС могут зависеть жизни людей;
- *Формализм* информационной системы – напрямую зависит от количества сотрудников, задействованных в проекте, чем выше их количество, тем более высокий показатель формализма требует проект;
- *Условия заказчика* – как правило, государственные структуры предъявляют более строгие требования к проекту;
- *Новизна разработки* – новые способы разработки предполагают повышение уровня формализма, требуя особых временных затрат на этапе подготовки стратегии проекта;
- *Удаленность сотрудников* - уровень формализма закономерно падает, если сотрудники работают в непосредственной близости и контакте друг с другом – это упрощает процесс обмена информацией.

Указанные факторы позволяют классифицировать модели ведения проектов, оценивая их «вес», а именно регламентированность рабочих процессов и общий формализм разработки. Показатели «веса» модели зависят от задокументированности рабочих процессов, оценка по этим критериям представлена в таблице 3.

Таблица 3 – Достоинства и недостатки моделей разработки по критерию «веса» рабочих процессов

| «Вес» | Модель | Достоинства | Недостатки |
|--------------|---------------------|--|---|
| Тяжелый | MSF RUP | <ul style="list-style-type: none"> - Заниженные требования к постоянному составу команды; - Не ограничены количество участвующих сотрудников и масштабы проекта; - Ориентированы на сотрудников средней квалификации; - Традиционно применяют для проектов, критичных к отказоустойчивости; - Упрощенное получение сертификации качества. | <ul style="list-style-type: none"> - Формализованный обмен информацией между сотрудниками; - Требует надстройки управления; - Значительно увеличиваются затраты на ведение документации; - Увеличенные сроки проектирования. |
| Легкий | Agile PSP TSP | <ul style="list-style-type: none"> - Основной акцент делается на разработке функциональных аспектов информационной системы; - Обмен информацией между сотрудниками лишен формализма; - Пониженные затраты на сопровождение, контроль, учет рисков. | <ul style="list-style-type: none"> - Требует квалифицированный кадровый состав для успешной реализации; - Сотрудникам необходимо располагаться локально, в непосредственной близости друг от друга; - Затрудненная процедура сертификации; - Используется для информационных систем низких рисков и малого масштаба; - За редким исключением неприменимы в проектах, разрабатываемых с нуля. |

В конце прошлого века были предложены новые, более совершенные стандарты качества, что в свою очередь привело к появлению нескольких

методологий, применяемых в данный момент, в числе которых: ISO/IEC 15004, представленный в России как «ГОСТ Р ИСО/МЭК 15004 «Информационные технологии. Оценка процессов», ISO 9001:2008, общепринятый стандарт ГОСТ и Capability Maturity Model.

Capability Maturity Model был разработан с целью создания эталонной модели проектирования крупных ИС, с высоким уровнем формализованных процессов.

ISO/IEC 15004 был предложен «Международной Организацией Стандартизации», считается эталонной моделью оценки качества процессов.

ISO 9001:2008 ориентирован на удовлетворение запросов конечного потребителя с помощью улучшения показателей управления качеством.

Применяемые в Российской Федерации стандарты *ГОСТ* используются для ступенчатых, последовательных методов создания информационных систем.

В табл. 4 указана возможность использования различных средств обеспечения качества ИС, использующих итеративный подход для разработки.

Таблица 4 - Возможность сертификации моделей итеративной разработки ПО

| Модель Стандарт | RUP | MSF for CMM | MSF for Agile | PSP/TSP | Agile |
|--------------------|-----------------|-------------------|-------------------|-----------------|-------|
| ISO 9000:2008 | + | + | - | - | - |
| CMM/CMMI | 3-ий уровень | 2-3-ий уровень | 2-3-ий уровень | 5-ый уровень | - |
| SPICE | + | + | + | + | + |
| ГОСТ | - | - | - | - | - |

где «-» означает невозможность сертификации; «+» - возможность сертификации.

Основная проблема использования ГОСТов и стандартов ISO 9000:2008 - это слабая формализация процессов при использовании гибкой методологии.

Наиболее эффективными средствами обеспечения качества для ИТ, использующих итеративную разработку, являются CMM ISO/IEC 15504 (SPICE), т.к. их использование не зависит от используемой модели разработки ИС.

В табл. 5 приводится сравнение моделей по показателям содержания, веса, ориентации на числовое разработчиков, использующимся стандартам для обеспечения качества, недостаткам и достоинствам.

Если размер команды достаточно велик (больше десяти разработчиков) и разрабатывает ИТ для разных отраслей либо систем разного уровня критичности – желательно осуществлять выбор для каждой разработки своего оптимального уровня формализации и модели разработки.

Разработка ИС связана с совокупностью рисков, которые влекут за собой возникновение неопределенности в процессе создания информационного продукта. Риски влияют на все без исключения аспекты деятельности, поэтому их обязательно необходимо учитывать для разработки надежной ИС.

Реакции на появление разных рисков и, следовательно, методы управления рисками будут меняться, ориентируясь на выбранную модель разработки ИС. В таблице 6 показаны методы управления рисками для самых распространенных моделей разработки информационных систем с итерационным циклом разработки - *RUP, Agile* и *MSF*.

Таблица 5 - Сравнение моделей ИС

| Модели Параметры | RUP | MSF | PSP/TSP | Agile | | | |
|------------------------|---|---|--|--|---|---|---|
| | | | | XP | Crystal Clear | FDD | Scrum |
| Содержание | абстрактный общий процесс, на базе которого организацией должен создаваться определенный специализированный процесс, который ориентирован на ее потребности | согласованный комплекс концепций, правил и моделей, которые описывают управление процессами и людьми при разработке решения; основывается практическим опытом Microsoft | Комплекс требований к компетенциям разработчика, ставка делается на самоуправляемые команды | комплекс практик: переработки кода, короткие релизы, коллективное владение кодом, разработка «тестами вперед», стандарты кода и постоянное присутствие заказчика | Итеративная инкрементная разработка; автоматическое регрессионное тестирование; состав документов определяют участники разработки | функционально-ориентированная разработка, оперирующая понятием функции или свойствами системы | итеративная разработка, фокусирующаяся на систематическом определении приоритетных задач, ориентируясь на бизнес цели |
| Команда | более 40-50 человек | группы от 3 до 10 человек | От 3 до 20 разработчиков | От 10 до 15 человек | до 6 человек | От 20 до 30 человек | От 5 до 9 человек |
| Вес | тяжелый | может быть как легким, так и тяжелым, | легкий | | легкий | | |
| Используемые стандарты | CMMI, ISO, SPICE | CMMI, ISO, SPICE | CMMI, SPICE | | CMMI, SPICE | | |
| Достоинства | 1) снижают основные риски разработчика и заказчика; 2) экономят ресурсы через автоматизацию регрессионного тестирования; 3) улучшают качество ПО через многократные проверки; 4) улучшают качество тестирования через пользование современными технологиями. | 1)систематизируют и структурируют информацию в форме базы знаний; 2)нестандартно подходят к процессам организационной структуры, распределения ответственности и принципам взаимодействия внутри команды | 1) лучше планируют время и бюджет; 2) управляют качеством продукта; 3) уменьшают время разработки; 4) регламентируют внутри команды принципы взаимодействия | 1) крайне низкие накладные расходы; 2) процессом может показываться исключительная эффективность | 1) максимальная простота в пользовании; 2) Нуждается в минимальных усилиях для внедрения; 3) основывается человеческими привычками; | предоставляет возможность планировать и предварительно проектировать, создавать детальный дизайн. | Дает возможность реализации большого объема функциональности без спецификаций на момент начала разработки |
| Недостатки | несогласованность решений, непродуктивные расходы ресурсов по переработке кода и для повторного решения типовых проблем при недостаточном уровне формализма | детально не описываются важнейшие роли пользователя и заказчика; не рассмотрены методы управления группой проектов | сложности в процессе учета рисков | расчитана для команды разработчиков с опытом, которая не разбита на несколько частей; проблемы при разрешении нетехнологических рисков | уступает по производительности XP; очень сложно предварительно сформулировать то, какие потребуются промежуточные продукты | недостаточно документации; значительно возрастает стоимость дальнейшего сопровождения продукта | не задаются технические практики; сложно достичь активного участия разработчиков и вовлечь в активное участие поставщика требований |

Таблица 6 – Методы для предотвращения наиболее распространенных рисков

| Модель Тип риска \ | <u>RUP</u> | <u>MSF</u> | <u>Agile</u> |
|--|---|---|--|
| Нарушения календарного планирования, срыв сроков | Перерасчет периода выполнения этапов работ | Планирование резерва времени | Выделение сверхурочных трудовых часов, увеличенное внимание к предварительному планированию |
| | <p>Ликвидировать или уменьшить данный риск можно через оценивание длительности конкретных работ либо всей разработки по нескольким методам:</p> <ul style="list-style-type: none"> - опрос экспертов является проведением интервью специалистов соответствующей квалификации; - мозговой штурм является проведением общего собрания, на котором специалистами по очереди высказывается свое мнение про необходимые временные затраты на конкретные типы работы. Не допускаются замечания и споры, все мнения записывают, группируют и оценивают в зависимости от типов и характеристик работ; - сбор данных является проведением по нескольким ИС сбора данных по отношению к масштабам недооценки размера разработки; - имитационное моделирование - моделирование и анализ неопределенности в оценках основных показателей разработки (денежные и временные затраты). | | |
| Изменение требований | учет увеличения временных затрат и трудоемкости при возможном росте требований, к примеру, на 50% (принятие риска) | переоценивание сроков разработки каждый раз, когда требования добавляют/изменяют (ликвидация риска) | подписание контракта с компенсацией затрат (переадресация риска заказчику) |
| Текущесть кадров | выявление причин, из-за которых людьми покидается компания. Для того, чтобы изучить мотивы ухода, важны сбор и анализ сведений про них. Прежде всего, это сведения про общее число уволившихся, долю сотрудников разных возрастных категорий, про работников с высокой и низкой квалификацией, а также про их стаж работы и образование | определение общих временных потерь: количество месяцев, необходимых для нового сотрудника, чтобы достичь одинакового уровня производительности, который был у работника, замененного им | <ul style="list-style-type: none"> - учет в оценках трудоемкости издержек для обучения персонала; - снижение количества потерь от текучести кадров через привлечение избыточного числа участников на начальном этапе |

| <u>Модель</u> <u>Тип риска</u> | <u>RUP</u> | <u>MSF</u> | <u>Agile</u> |
|---|---|--|---|
| Нарушение спецификаций | Политика ликвидации риска является подписанием договора заказчика и компании с описанием выходных и входных условий, оцениванием рисков независимыми экспертами | | |
| Низкая производительность | улучшение организации процессов производства и труда - повышение норм труда и расширения зон обслуживания; снижение количества сотрудников, которые не выполняют нормы; упрощение структуры управления; рост уровня специализации | структурные изменения - изменения удельных весов отдельных работников, ориентируясь на трудоемкость выполняемой работы | повышение технического уровня производства - внедрение инновационного оборудования и программного обеспечения |
| От руководства компании недостаточно внимания к проекту | Проведение промежуточных отчетных собраний с приглашением заказчика | Промежуточное оценивание руководством компании прогресса разработки ИС с формированием отчетности перед заказчиком | Составление договора заказчика и компании с поэтапной сдачей ИС |
| Отсутствует мотивация сотрудников компаний | Проведение корпоративных собраний, тренингов, внедрение методики team building | Премирование сотрудников, ориентируясь на успешность выполнения задач | Распространение среди персонала акций компании, выделение % от прибыли с разработки разработчикам |

Совершенствование информационных технологий происходит каждый день, из-за чего также повышается качество управления рисками. Некоторые методики естественно стареют, другие появляются и совершенствуются. В итоге на рынке программ оценивания рисков остается только несколько лидеров, превосходящих другие аналоги, которые редко обновляются или неэффективны, а в самих методиках появляются различия, которыми и основываются все недостатки и достоинства программных комплексов.

Сначала следует рассмотреть известные модели: Cramm, Octave, ГРИФ, RiskWatch на базе работ, предназначенных для оценивания рисков внедрения ИТ. На базе этих моделей с соответствующим названием создаются программные продукты. В таблице 7 проведены обзор и сравнение данных методов. Важно

отметить, что в них не учтена специфика модели взаимодействия, свойственная облачным средам, а конкретно, - удалённый доступ к предоставляемым сервисам. Вдобавок, у данных методов имеется только качественная оценка риска.

Таблица 7 - Обзор методов оценивания рисков использования ИТ

| Название методики | Смысл метода | Этапы оценки | Достоинства | Недостатки |
|-------------------|---|---|--|---|
| Octave | Анализ риска производят только сотрудники предприятия без помощи консультантов внешних компаний | 1. Разработка профиля угроз, связанных с активом. 2. Идентификация инфраструктурных уязвимостей. 3. Разработка стратегии и планов безопасности. | Позволяется всесторонняя оценка всех последствий для бизнеса по причине возможных инцидентов в области ИБ, а также разработка контрмер. | Выполняется только оценивание ожидаемого ущерба, без оценки его вероятности. |
| Cramm | Универсальный инструмент, который позволяет, кроме анализа рисков, обследовать ИС и осуществлять выпуск сопроводительной документации на каждом этапе проведения анализа; разработка политики безопасности и плана обеспечения непрерывности бизнеса. | 1. Идентификация и определение ценности защищаемых ресурсов. 2. Идентификация и оценка угроз в сфере ИБ, поиск и оценивание уязвимостей защищаемой системы. 3. Генерирование вариантов мер воздействия на выявленные риски. | Имеется большая база, в которой содержится описание примерно 1000 примеров реализации подсистемы защиты компьютерных систем; сочетается количественные и качественные методы анализа [49]. | Требуется высокая квалификация аудитора и специальная подготовка; аудит – является процессом достаточно трудоемким; инструментарием программы генерируется большое число бумажной документации; отсутствует возможность создания собственных шаблонов отчетов и модификации имеющейся; отсутствует возможность внесения знаний и дополнений в базу. |

| Название методики | Смысл метода | Этапы оценки | Достоинства | Недостатки |
|-------------------|--|---|--|---|
| RiskWatch | Риски информационной и физической безопасности сети рассматриваются совместно. Для оценивания и управления рисками. В виде критериев пользуются ROI и ALE. | 1. Определяется предмет исследования. 2. Вводятся данные, описывающие определенные характеристики системы. 3. Количественная оценка (здесь рассчитывают профиль рисков, а также выбирают меры обеспечения безопасности). 4. Генерируются отчеты. | Точное количественное оценивание соотношения потерь от угроз с затратами на создание системы защиты. | Подходит лишь для анализа рисков на программно-техническом уровне защиты, не учтены административные и организационные факторы; полученной оценкой рисков (математическое ожидание потерь) не дается понимание риска с системной позиции. |
| ГРИФ | Анализируется уровень защиты ресурсов, проводится оценка возможного ущерба от реализации угроз ИБ и оказывается помощь в управлении рисками, с выбором контрмер. | Этапы, на которых менеджером вводится: 1) список ресурсов организации, виды данных; 2) ущерб по каждой группе сведений; 3) доступ у пользователей к ресурсам; 4) средства для защиты; 5) отвечает на определенные вопросы системы. | Подготавливается подробный отчет про уровень риска каждого ресурса ИС компании, уточняются причины возникновения риска с анализом выявленных уязвимостей и оцениванием экономического эффекта от каждой контрмеры. | Сложность во внутренних алгоритмах |

2 РАЗРАБОТКА АЛГОРИТМОВ ВНЕДРЕНИЯ МОБИЛЬНОЙ КОРПОРАТИВНОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ

2.1 Проблемы корпоративной мобильности предприятия

В современном мире многие организации не успевают за темпом технического развития, что в итоге приводит если не к ликвидации, то к существенному снижению потенциальной прибыли предприятия.

Информационная система позволяет предоставить организации необходимый инструментарий для создания на ее базе корпоративных мобильных решений. К числу этих инструментов можно отнести виртуальные сети, беспроводные технологии и системы виртуализации. Все они направлены на решение ряда проблем, таких как:

- система организации должна учитывать принципы BYOD;
- предоставляемый сервис должен соответствовать ожиданиям пользователя в аспектах производительности и доступности;
- время получения данных не должно влиять на реальный опыт использования системы пользователем.

Иными словами, организация должна предоставлять end-to-end услуги, что в свою очередь требует определенных финансовых и временных затрат на модернизацию существующей и работающей структуры. Это позволит пользователям работать с данными там, где это удобно, увеличит общую производительность и конкурентоспособность организации [9].

При разработке мобильной стратегии ориентируются на 3 показателя:

- данные;
- устройства;
- пользователи.

В качестве доступных сценариев предоставления мобильных сервисов чаще всего рассматривают следующие:

- Конфиденциальность корпоративной информации;
- Администрирование мобильных устройств;
- Свобода в выборе устройств и операционных систем.

Пользовательский опыт и разработка мобильных приложений основываются на требованиях, предъявляемых к работе с корпоративными данными. Следует оценивать риски при синхронизации, хранении, обработке, аутентичности информации, внезапной компрометации или утере данных. Эти факторы непосредственно влияют на безопасность, опыт пользователя и дизайн мобильного приложения.

Разработка мобильного приложения неизбежно связана с рядом проблем, таких как:

- управление потреблением контента и его распространением;
- определение определенного объема некритичной к утечке информации, не представляющей особой угрозы для компании;
- корректное масштабирование элементов интерфейса и учет специальных требований сотрудников с дефектами здоровья;
- стоимость разработки архитектуры и уровень профессиональных компетенций сотрудников, непосредственно участвующих в процессе разработки.

Решение ряда проблем, растущих вместе со сложностью системы, увеличивает стоимость ее разработки и требования, предъявляемые пользователями.

Помимо этого, остаются актуальными проблемы кражи оборудования, внедрения вредоносного программного обеспечения, утечки данных в результате действий пользователей. Мобильность предполагает, что организация также будет вынуждена учитывать следующие факторы:

- короткий жизненный цикл устройства;
- отсутствие контроля и аудита используемых сотрудниками устройств;
- невозможность контролировать направление передачи данных пользователями;
- отсутствие определенного стандарта устройств;
- устройство может не поддерживать программное обеспечение, обеспечивающее контроль над безопасностью;
- системы могут сильно отличаться друг от друга в методах реализации

внешней и внутренней безопасности;

- пользователю необходимо подтвердить законность проведения обмена данными, а значит пройти аутентификацию.

Внедрение сложной корпоративной мобильной информационной системы невозможно без предварительного тестирования и решения возможных проблем доступа к внедряемой системе.

2.2 Методы и методики выбора внедряемых облачных сервисов в организации

Методику поддержки принятия решений в процессе выбора облачных ИТ-сервисов для внедрения в организации предоставили на рисунке 8.

Первый этап «Определение затрат и выгод» заключается в формировании набора приложений для оценивания и определения затрат и выгод внедрения облачных сервисов. На этапе по определению высокоуровневых требований бизнеса должны быть выявлены:

1. Функции бизнеса;
2. Основные причины, которые провоцируют бизнес на внедрение облачных сервисов (стратегические цели);
3. Облачные сервисы, которыми поддерживались бы бизнес-процессы;
4. Значимые требования законодательства;
5. Физическое размещение систем, которые обеспечивают предоставление услуг (на территории организации, не на её территории, в конкретной географической точке) и кто будет ответственным за оказание услуг.

Далее определяют, какой для предприятия необходим тип облачной модели (SaaS, PaaS, IaaS), а также какая модель размещения облака (публичная, частная, общественная, гибридная) лучше всего подходит для организации.

Следующим этапом является определение базовой/стартовой модели облачного сервиса с позиции риска. Здесь определяют области риска, которые должны приниматься в расчет, а также меры снижения риска в изучаемых областях до уровня, который приемлем со стороны предприятия.

Примерами мер снижения риска являются:

- шифрование клиентом данных для защиты их от несанкционированного доступа сотрудниками облачного провайдера;
- разработка стратегии по возврату в исходное состояние при провале бизнеса облачного провайдера;
- резервная копия данных/отслеживание клиентом аудита на своей территории при потере доступа к облачному сервису;
- полно или ясно сформулированные SLA (Service Level Agreement), в которых включен пункт права на аудит;
- составляется и осуществляется внутренний план по восстановлению после аварии.

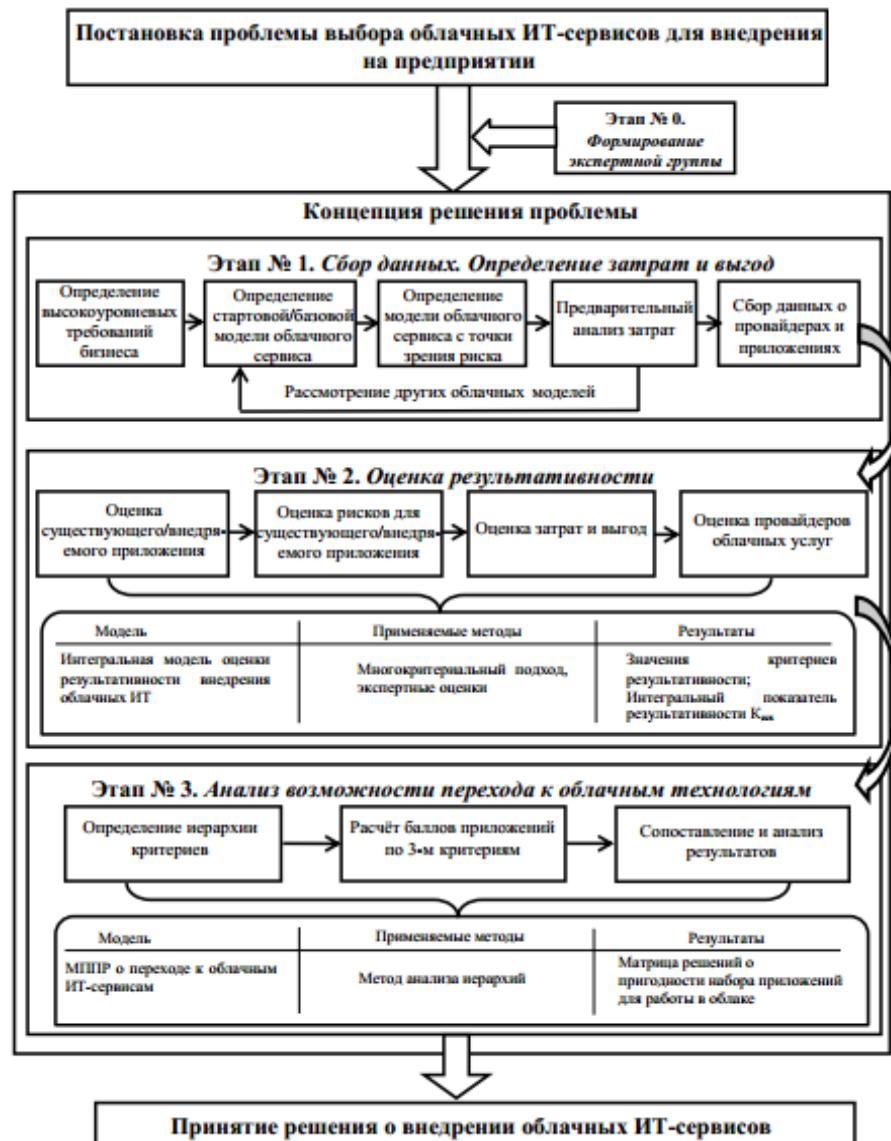


Рисунок 12 - Методика поддержки принятия решений при выборе облачных ИТ-сервисов для внедрения на предприятии

В предварительный анализ затрат включаются:

1. Затраты миграции на облачную модель с существующей модели (процесс разработки приложений, переформатирование данных в соответствии с форматами провайдера SaaS, настройка объединенного управления контролем доступа и идентичности, реализация процессов управления облаком).
2. Затраты по работе с облачной моделью (выплата провайдеру, поддержка и лицензия, передача данных).
3. Постоянные и разовые затраты по снижению рисков (инструменты шифрования данных, тестирование и планирование стратегий возврата в первичное состояние, обслуживание, независимого резервного копирования от провайдера).

При изучении других облачных моделей определяет: уменьшатся ли расходы, если изменится модель размещения/предоставления облачных сервисов; будет ли возможность пользоваться частным, общественным или гибридным облаком и отказа от некоторых мер безопасности, которые важны в публичном облаке; есть ли возможность сокращения расходов по снижению риска, который связан с привязкой к конкретному производителю через пользование моделью IaaS или PaaS вместо SaaS.

Вторым этапом «Оценка результативности» является оценивание всех существующих приложений и/либо такого приложения, которое необходимо внедрить на предприятии согласно функциональным и юридическим требованиям бизнеса (установленным на 1 этапе).

При оценивании рисков для внедряемого /существующего приложения определяют:

1. Области, в которых необходимо снизить риск из-за превышения приемлемого для предприятия уровня.
2. Меры, помогающие в снижении риска до приемлемого уровня (к примеру, пользоваться частным облаком, чтобы не приходилось делиться площадкой с другими компаниями, провести оценивание поставщика, а также наличие у него сертификатов и т. д.).

3. Сравнение двух подобных. Для этого проводится анализ области риска, который существует для текущей технологии, с целью убедиться, что в оценивании будущего и существующего состояния учитывается одно и то же.

При оценке выгод и затрат определяют:

1. Текущие затраты по эксплуатации/обслуживанию.
2. Скрытые затраты, в том числе затраты по снижению рисков.
3. Нематериальные и материальные выгоды.

После проведения анализа выгод и затрат, выполняют расчет критериев и интегрального показателя «Результативность облачного ИТ-сервиса» в соответствии с предложенной системой критериев оценивания преимуществ и интегральной модели, основываясь многокритериальным подходом принятия решений. При проведении рассчёта критерия K_{ecs} и показателей рисков и эффективности требуется подключение к работе экспертов, финансового отдела и использование корпоративных стандартов.

Третий этап «Анализ возможности перехода к облачным технологиям» заключается в определении иерархии критериев. Далее выполняется расчёт баллов приложений в соответствии с тремя критериями предложенной модели. По результатам расчетов составляют матрицу принятия решений про возможность перехода в облачную среду приложений и принимают решение выбора облачных ИТ-сервисов для внедрения в компании.

В предлагаемой методике по поддержке принятия решений выбора облачных ИТ-сервисов для внедрения возможно выделение следующей последовательности действий в формате блок-схемы (рис. 13), по которой выполняется процесс оценивания и отбора приложений для внедрения в организации с использованием предложенных моделей.

После того, как выбраны подходящие приложения для перехода в облачную среду, в соответствии с разработанной методикой создают проекты для внедрения. При ограниченной бюджете предприятия, на базе этих проектов предложен просчет показателей эффективности - NPV (в виде лучшего критерия, который указывает правильные инвестиционные решения) и выбор проектов с наибольшим

комплексным NPV.

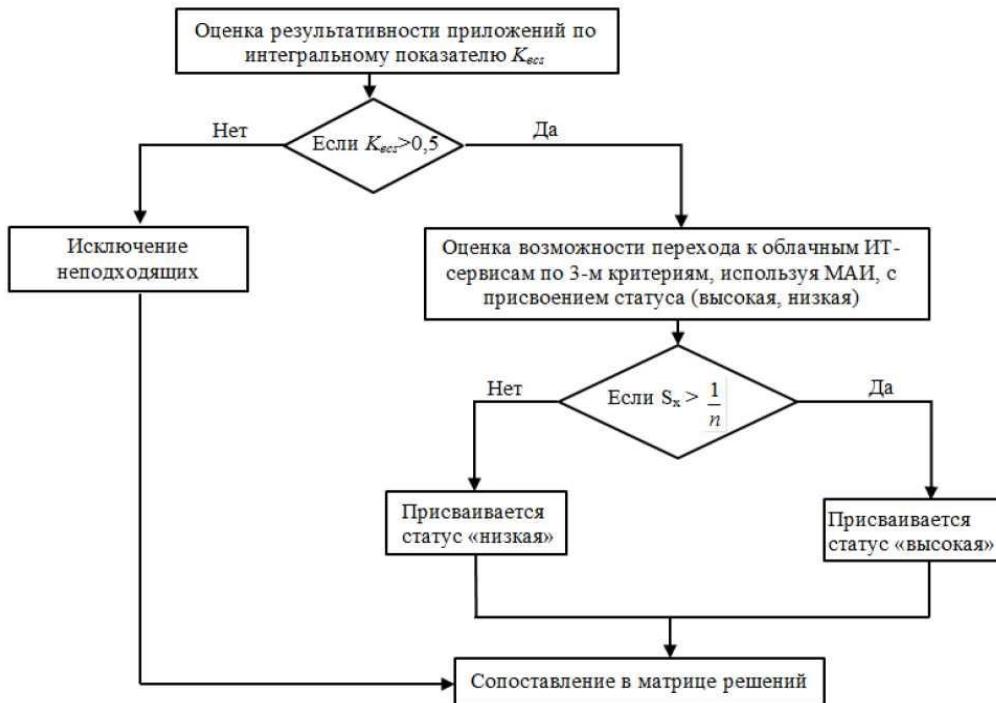


Рисунок 13 - Схема оценивания и отбора облачных ИТ-сервисов для внедрения в организации с использованием предложенными моделями поддержки принятия решений

Здесь предложена модель выбора инвестиционных программ в облачные технологии, основываясь методами математического программирования. Для этого требуется уже сформированный проект и перечень информации:

- численные показатели для хорошо оцениваемых и прямых выгод;
- расходы по внедрению облачных ИТ;
- чистые прибыли и расходы для каждого года.

2.3 Модель и метод внедряемых мобильных корпоративных информационных систем

Для того, чтобы лучше понимать все выявленные в литературе преимущества, расходы, выгоды и риски от внедрения облачных сервисов, следует привести их в таблицах 8, 9 и 10. Главные моменты для оценивания эффективности от внедрения представлены в таблице 11. Основываясь этими обоснованными

сформированными данными, создали систему критериев и показателей оценивания результативности пользования облачными ИТ-сервисами, а также иерархию критериев в методике для поддержки принятия решения (МППР) при переходе к облачным ИТ-сервисам.

Таблица 8 - Преимущества облачных сервисов

| Вид преимущества | Объяснение преимущества облачных сервисов |
|---|---|
| Материальные преимущества. Прямые, вычисляемые | |
| Понижение затрат | <p>Затраты на ИТ трансформируются из капитальных в операционные, так как облачным провайдером предоставляется необходимая инфраструктура в формате части комплексной услуги.</p> <p>Вместе с этим, облачные сервисы способны на снижение капитальных и операционных затрат в сферах:</p> <ul style="list-style-type: none"> - администрирования ИТ-системы и технической поддержки (меньшее количество персонала и часов работы); - приобретения и поддержки лицензий; - сопровождения (установки патчей и обновлений, замен версий); - хостинга (физического здания, электропитания, и т.д.) |
| Увеличение скорости (гибкости) | <p>Через более оперативное предоставление сервисов происходит ускорение:</p> <ul style="list-style-type: none"> - внедрения и развертывания приложений (с помощью SaaS); - разработки и тестирования приложений (с помощью PaaS). <p>Скоростью оказывается помочь в снижении расходов по подключению новых пользователей (масштабированию) и нового функционала. Преимуществом облачных провайдеров является способность более быстрого внедрения технических инноваций, нежели традиционными ИТ-службами.</p> |
| Рост производительности пользовательской работы | Снижением затрат и периода обработки инцидентов и изменений провоцируется повышение производительности |
| Надежность | Снижение периода обработки инцидентов. В распоряжении облачных провайдеров находятся избыточные площадки, которыми можно эффективно воспользоваться для восстановления работоспособности и непрерывности бизнеса |
| Улучшение в соблюдении требований безопасности | Облачными провайдерами могут быть предложены мощные механизмы для обеспечения безопасности. |
| Материальные преимущества. Прямые и сложно-вычисляемые | |
| Оптимизация пользования ресурсами | Снижение простоев вычислительных систем, так как предприятия пользуются только теми вычислительными ресурсами, которые нужны |
| Производительность | Максимальная производительность ИТ-систем обеспечивается через использование современными средствами мониторинга. |
| Масштабирование | Благодаря обеспечению вычислительных ресурсов по требованию, расходы на планирование мощностей и предоставление пользователям доступа становятся ненужным. |

| Вид преимущества | Объяснение преимущества облачных сервисов |
|---|---|
| Удовлетворенность работников организации | Снижением времени на отклик на запросы и инциденты может повышаться удовлетворенность работников компании. |
| Нематериальные преимущества | |
| Доступ к умениям и навыкам профессионалов | Заказчиками получаются специалисты высшей квалификации, без понесения расходов, которые связаны с процессами их найма (поиска, окладов, премий, обучения и т. д.) |
| Удовлетворенность работников компании | Высокое быстродействие и мобильность являются повышением удовлетворенности работников компании. |
| Удовлетворенность клиентов организации | взаимодействие предприятия и его клиентов улучшается, время реакции на запросы клиентов сокращаются с помощью эффективного пользования облачными приложениями. |
| Фокусировка на ключевых задачах | Возможность направления ИТ-ресурсов на поддержку главных бизнес-функций |
| Перенос риска | Определенными рисками (к примеру, бреши в системе защиты, восстановление из-за сбоев) допускается передача облачному провайдеру. |
| Новые возможности для бизнеса | Облачное приложение (SaaS) может выступать как критически важный элемент при формировании нового бизнеса либо в процессе выхода на новый рынок. К примеру, способность быстрого реагирования на рыночные изменения через ускоренный выпуск нового продукта либо резкое изменение объемов выпуска. |

Таблица 9 - Расходы на облачные сервисы

| Расходы | Пояснение расходов на облачные сервисы |
|---------------------------------------|---|
| | Расходы при запуске |
| Техническая готовность инфраструктуры | Для того, чтобы выполнить новые требования к пропускной способности каналов /сети доступа в Интернет для сетевой инфраструктуры необходимы определенные инвестиции. Зачастую, кроме этого, следует модернизировать другие компоненты инфраструктуры для того, чтобы обеспечить их интеграцию с облачными сервисами. |
| Переход к облачным сервисам | Для проекта перехода на облачные сервисы может потребоваться привлечение услуг профессионалов. |
| Интеграция | Интеграция облачных и локальных сервисов может нуждаться в привлечении профессиональных услуг. |
| Конфигурирование /настройка | При использовании SaaS-приложениями необходима их конфигурация под требования предприятия. |
| Обучение | Сотрудники ИТ-департамента нуждаются в обучении сотрудничеству с поставщиками сервисов и управлению обменом информацией с поставщиками, а конечные пользователи - в обучении работать с новыми приложениями. |

| Расходы | Пояснение расходов на облачные сервисы |
|--|--|
| Организационные изменения | Реорганизация процессов управления изменениями и взаимоотношениями с поставщиками, мониторинг пользования ресурсами, предоставление возможности доступа пользователей, внутренний аудит. |
| Операционные расходы | |
| Абонплата | Оговоренный размер суммы, периодически (ежегодно, ежеквартально или ежемесячно) выплачиваемый за использование облачных сервисов. |
| Управление изменениями | Расходы, которые относятся к управлению изменениями, и иные расходы, которые возникают из-за запросов каких-либо изменений. Если в договоре не предусматривается иное, то некоторыми поставщиками может браться плата за наращивание или сокращение вычислительных ресурсов. |
| Сотрудничество с провайдером | Расходы для мониторинга работы облачных провайдеров, управление договорами, пользование и мониторинг SLA (соглашение про уровень обслуживания), а также для иной деятельности контроля и оценивания предоставляемых услуг. |
| Координация облачных сервисов | Если предприятием используется более 1-го сервиса, требуется создание группы для координации, которая будет обеспечивать бесконфликтную работу и интеграцию сервисов. |
| Администрирование и поддержка пользователей | Часть расходов уже состоит в абонентской плате, а часть - остается у фирмы. |
| Понижение рисков | Требуется применение контрмер против новых рисков, которые создают облачные вычисления. |
| Расходы на прекращение пользования облачными сервисами у определенного провайдера | |
| Переход к другому провайдеру либо возвращение к собственной инфраструктуре | <p>Если пользование облачными сервисами становится невыгодным (из-за изменения законодательного контроля или экономических условий работы с поставщиком), организации необходимо поменять поставщика услуг либо осуществить возврат к собственной инфраструктуре.</p> <p>Расходы:</p> <ul style="list-style-type: none"> - извлечь данные из облака и проверить их корректность и полноту; - стереть или уничтожить данные в облачных системах обработки и хранения информации; - получить и конфигурировать системы для замены облачных сервисов (собственные либо другого поставщика облачных услуг); - оплатить штрафы за досрочное расторжение контракта; - выделить штат ИТ-специалистов либо нанять дополнительных специалистов для того, чтобы поддержать возвращенные сервисы; - выделить или приобрести физические ресурсы для того, чтобы разместить возвращенные сервисы. |

Таблица 10 - Риски при использовании облачных сервисов

| Виды рисков | Пояснения рисков при использовании облачных ИТ |
|--|---|
| Ограничение возможностей по настройке приложений | Невозможность перенастроить облачное приложение при любой смене бизнес-процесса, потому что данная модификация производится за отдельную плату или невозможна |
| Соблюдение законных требований | Расхождение операции и действующих правовых норм (причем это своевременно не замечается). Проблема поясняется повсеместной доступностью облачных сервисов и эволюцией регулирующих требований. |
| Безопасность | Облачные вычисления провоцируют традиционные и новые риски в сфере безопасности, которыми учитывается специфика облачных ИТ. Провайдеру и клиенту необходимо учитывать данные риски и принимать для их снижения адекватные меры. |
| Несовместимость | Несовместимость облачных сервисов и имеющейся ИТ-инфраструктуры либо специфических систем, которые необходимо интегрировать с сервисами. |
| Бесперебойная работа | Отсутствует гарантия времени бесперебойной работы, оговоренная в контракте. Кроме этого, сбои могут вызывать другие факторы. |
| Переплата по схеме pay-as-you-go | Привлеченные дополнительные ресурсы могут быть подключены по окончанию пикового спроса. |
| Привязка к поставщику или аппаратной платформе | Привязка заказчика к конкретной облачной технологии либо поставщику облачных сервисов, невозможность перехода на другую платформу. |
| Утрата внутренних компетенций | Утрачиваются основные внутренние навыки в сфере ИТ. Вероятно, они могли бы выступать, как один из моментов стратегического дифференцирования организации. |
| Производительность | Понижение производительности систем, когда провайдер пользуется моделью разделения ресурсов в процессе предоставления сервисов для нескольких заказчиков и при планировании мощностей допускает ошибку. Также скорость доступа в Интернет может отражаться на производительности. |
| Консьюмеризация облака | Бизнес-единицами могут заказываться облачные сервисы без участия ИТ-департамента по причине широкого распространения простых облачных сервисов. |

Таблица 11 - Ключевые моменты для оценки эффективности от внедрения

| Моменты при оценке | Пояснения ключевых моментов |
|--|---|
| Облако подходит не для всего и не всем | Критически важным является процесс выбора типа облачного сервиса и способа управления. Первое место занимает стратегическое оценивание затрат, преимуществ, рисков. |

| Моменты при оценке | Пояснения ключевых моментов |
|---|--|
| Учет всей совокупности взаимосвязанных потребностей компаний | Облачные вычисления оказывают влияние на большое количество разных элементов (персонал ИТ-отдела, инструменты для поддержки ИТ, ИТ-процессы и бизнес-процессы), их необходимо учитывать при проведении анализа. |
| Учет многочисленных скрытых затрат, которые скрываются в расценках провайдера | Провайдером не запрашивается никакая предварительная плата за предоставление своих сервисов, но для миграции существующей системы в облако необходимы значительные затраты. |
| Оценивание эффекта не всегда должно быть сложным | Простым, но эффективным расчетом ROI или ТСО позволяет обеспечить для компании поддержку процесса принятия решения по инвестициям и соотношение ожидаемых затрат с пользой. Переусложненным расчетом может затрудняться понимание причин, по которым принято было какое-либо решение. |
| Подходы и политики | Подход компании к процессам подготовки и оценки бизнес-кейсов (фокусировка на нематериальных и материальных преимуществах?) и стадия развития организации (это зрелый бизнес или стартап?) |

Рассмотрим три альтернативы заключения договоров с фирмами на поставку программных продуктов для внедрения их на предприятии:

- 1) компания для поставки "1С Предприятие" условно назовем для расчетов ("КП");
- 2) компания "Галактика Business Suite" условно назовем для расчетов ("А");
- 3) компания "MicrosoftAxapta" условно назовем для расчетов ("А");

В задаче принятия решения по выбору поставщика консервов учитываются как такие факторы:

- надежность (Н);
- качество программного продукта (К);
- цена программного продукта (Ц)
- уровень развития инфраструктуры поставок и известность на рынке программных продуктов(Р);

Приоритеты критериев представлены матрицей попарных сравнений:

| | H | K | Ц | P |
|---|------|------|------|---|
| H | 1 | 2 | 3 | 4 |
| K | 0,5 | 1 | 2 | 4 |
| Ц | 0,33 | 0,5 | 1 | 3 |
| P | 0,25 | 0,25 | 0,33 | 1 |

При объявлении тендера фирмы подали документы о своем состоянии, анализ которых приведен в табл. 12:

Таблица 12 - Информация о производителях ПО

| Фирма | Обозначение | Надежность | Качество сырья | Цена | Уровень развития |
|-------|-------------|----------------------------|----------------|------------|---|
| «КП» | КП | были проблемы с поставками | Высокое | Высокая | На рынке неизвестна |
| «А» | А | проблем не было | Среднее | Приемлемая | Имеет свой сегмент на рынке, инф-ра развита |
| «М» | М | проблем не было | Ниже среднего | Низкая | Имеет свой сегмент на рынке, инф-ра развита |

Результаты попарного сравнения альтернатив по перечисленным факторам, а также попарного сравнения критериев по степени их важности приведенные ниже.

Альтернатива КП – заключить договор с поставщиком

Альтернатива А – заключить договор с поставщиком

Альтернатива М – заключить договор с поставщиком

Сравнение альтернатив по критериям:

| | КП | А | М |
|----|----|-----|-----|
| КП | 1 | 0,2 | 0,2 |
| А | 5 | 1 | 1 |
| М | 5 | 1 | 1 |

| | КП | А | М |
|----|------|-----|---|
| КП | 1 | 4 | 5 |
| А | 0,25 | 1 | 2 |
| М | 0,2 | 0,5 | 1 |

| | КП | А | М |
|----|-------|----|-----|
| КП | 1 | 8 | 2 |
| А | 0,125 | 1 | 0,1 |
| М | 0,5 | 10 | 1 |

| | КП | А | М |
|----|------|-----|---|
| КП | 1 | 2 | 3 |
| А | 0,5 | 1 | 2 |
| М | 0,33 | 0,5 | 1 |

Проведем расчет весов критериев и альтернатив.

Критерии

| Критерии | H | K | Ц | P | Сравнение геометрическое | Веса W |
|----------|------|------|------|---|--------------------------|--------|
| H | 1 | 2 | 3 | 4 | 2,88 | 0,52 |
| K | 0,5 | 1 | 2 | 4 | 1,59 | 0,29 |
| Ц | 0,33 | 0,5 | 1 | 3 | 0,79 | 0,14 |
| P | 0,25 | 0,25 | 0,33 | 1 | 0,27 | 0,05 |
| Сумма | | | | | 5,54 | 1,00 |

Фактор надежности (H)

| Критерии | КП | A | M | Сравнение геометрическое | Веса W |
|----------|----|-----|-----|--------------------------|--------|
| КП | 1 | 0,2 | 0,2 | 0,34 | 0,1 |
| A | 5 | 1 | 1 | 1,71 | 0,45 |
| M | 5 | 1 | 1 | 1,71 | 0,45 |
| Сумма | | | | 3,76 | 1,00 |

Фактор качества (K)

| Критерии | КП | A | M | Сравнение геометрическое | Веса W |
|----------|------|-----|---|--------------------------|--------|
| КП | 1 | 4 | 5 | 2,71 | 0,68 |
| A | 0,25 | 1 | 2 | 0,79 | 0,20 |
| M | 0,2 | 0,5 | 1 | 0,46 | 0,12 |
| Сумма | | | | 3,96 | 1,00 |

Фактор цены (Ц)

| Критерии | КП | A | M | Сравнение геометрическое | Веса W |
|----------|-------|----|-----|--------------------------|--------|
| КП | 1 | 8 | 2 | 2,52 | 0,57 |
| A | 0,125 | 1 | 0,1 | 0,23 | 0,05 |
| M | 0,5 | 10 | 1 | 1,71 | 0,38 |
| Сумма | | | | 4,46 | 1,00 |

Фактор развития (P)

| Критерии | КП | A | M | Сравнение геометрическое | Веса W |
|----------|------|-----|---|--------------------------|--------|
| КП | 1 | 2 | 3 | 1,82 | 0,54 |
| A | 0,5 | 1 | 2 | 1,00 | 0,30 |
| M | 0,33 | 0,5 | 1 | 0,55 | 0,16 |
| Сумма | | | | 3,37 | 1,00 |

Конечные оценки альтернатив рассчитываем путем перемножения двух матриц: первой, составленной с упорядоченных оценок альтернатив и второй, которая является весовыми коэффициентами оценок.

$$W_{\text{цель}}^A [W_H W_K W_\Delta W_P] \times [W_{\text{крит}}] = \begin{bmatrix} 0,1 & 0,68 & 0,57 & 0,54 \\ 0,45 & 0,2 & 0,05 & 0,3 \\ 0,45 & 0,12 & 0,38 & 0,16 \end{bmatrix} \times \begin{bmatrix} 0,52 \\ 0,29 \\ 0,14 \\ 0,05 \end{bmatrix} = \begin{bmatrix} 0,356 \\ 0,314 \\ 0,33 \end{bmatrix}$$

Из проведенного анализа видно, что преимущество для заключения контракта по поставке программного продукта имеет компания «КП».

Рассчитаем индексы согласованности матрицы парных сравнений AW .

$$AW_{\text{крит}} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 0,5 & 1 & 2 & 4 \\ 0,33 & 0,5 & 1 & 3 \\ 0,25 & 0,25 & 0,33 & 1 \end{bmatrix} \times \begin{bmatrix} 0,52 \\ 0,29 \\ 0,14 \\ 0,05 \end{bmatrix} = \begin{bmatrix} 1,72 \\ 1,03 \\ 0,61 \\ 0,3 \end{bmatrix}$$

Проведенный анализ согласованности парных сравнений показал в целом удовлетворительную оценку согласованности, кроме превышения (11%) сравнений критериев оценки.

Расчет индекса согласованности матриц парных сравнений

| Критерии | W | AW | λ |
|----------------------------|-------|--------------------|------------------------|
| H | 0,52 | 1,72 | 3,31 |
| K | 0,29 | 1,03 | 3,55 |
| Ц | 0,14 | 0,61 | 4,33 |
| P | 0,05 | 0,30 | 5,97 |
| Сумма | 1,000 | 3,66 | $\lambda_{\max} = 4,3$ |
| Индекс согласованности J | 0,1 | $= (4,3-4)/(4-1)$ | $n=4$ |
| Эталонное значение J | 0,9 | | $n=4$ |
| Отношение согласованности | | 0,1/0,9*100% = 11% | |

Фактор надежности

| Фактор | W | AW | λ |
|----------------------------|-------|------------------|-----------|
| КП | 0,1 | 0,28 | 2,80 |
| A | 0,45 | 1,4 | 3,11 |
| M | 0,45 | 1,4 | 3,11 |
| Сумма | 1,000 | | |
| | | λ_{\max} | 3,001 |
| Индекс согласованности J | | | 0,0001 |
| Эталонное значение J | | | 0,580 |
| Отношение согласованности | | | 0,048 % |

Фактор качества

| Фактор | W | AW | λ |
|----------------------------|-------|------------------|-----------|
| КП | 0,68 | 2,08 | 3,06 |
| А | 0,2 | 0,61 | 3,05 |
| М | 0,12 | 0,356 | 2,97 |
| Сумма | 1,000 | | |
| | | λ_{\max} | 3,03 |
| Индекс согласованности J | | | 0,01 |
| Эталонное значение J | | | 0,580 |
| Отношение согласованности | | | 2,2 % |

Фактор цены

| Фактор | W | AW | λ |
|----------------------------|-------|------------------|-----------|
| КП | 0,57 | 1,73 | 3,04 |
| А | 0,05 | 0,16 | 3,19 |
| М | 0,38 | 1,17 | 3,07 |
| Сумма | 1,000 | | |
| | | λ_{\max} | 3,1 |
| Индекс согласованности J | | | 0,05 |
| Эталонное значение J | | | 0,580 |
| Отношение согласованности | | | 8,2 % |

Фактор развития

| Фактор | W | AW | λ |
|----------------------------|-------|------------------|-----------|
| КП | 0,54 | 1,62 | 3,00 |
| А | 0,30 | 0,89 | 2,97 |
| М | 0,16 | 0,49 | 3,05 |
| Сумма | 1,000 | | |
| | | λ_{\max} | 3,01 |
| Индекс согласованности J | | | 0,0001 |
| Эталонное значение J | | | 0,580 |
| Отношение согласованности | | | 0,5 % |

На рисунках 14-16 представлен интерфейс программной реализации.

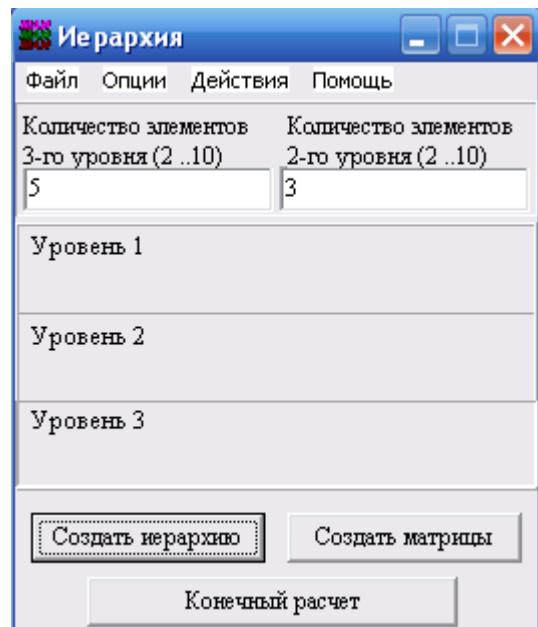


Рисунок 14 - Главное окно программы

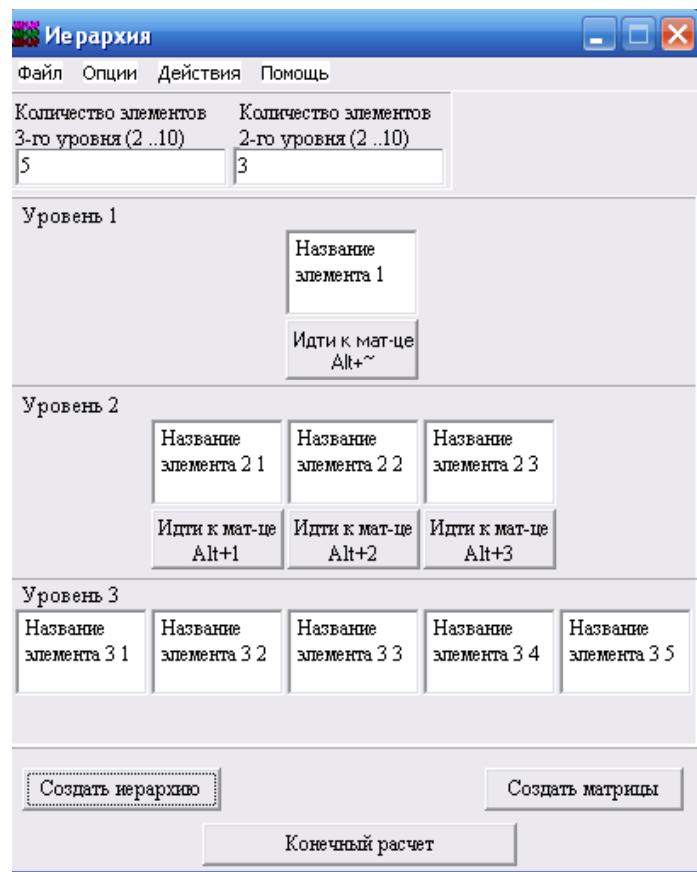


Рисунок 15 - Задание иерархии

| Критерий- Название элемента 2 3 | Название элемента | Вектор Приоритетов |
|---------------------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|--------------------|
| Название элемента 3 1 | 1 | | | | | | |
| Название элемента 3 2 | | 1 | | | | | |
| Название элемента 3 3 | | | 1 | | | | |
| Название элемента 3 4 | | | | 1 | | | |
| Название элемента 3 5 | | | | | 1 | | |

Lmax= ИС= ОС=

Вычисление Круговая диаграмма Столбчатая диаграмма

Рисунок 16 - Заполнение матрицы

2.4 Технологии тестирования и отладки программного продукта

2.4.1 Разработка и тестирование интерфейса приложений

При разработке интерфейса мобильных приложений необходимо учитывать ряд отличий, специфичных для данной платформы, таких как:

- удобство доступа до элементов интерфейса;
- размер экрана, его разрешающая способность;
- размеры и стили шрифтов;
- удобство расположения информации на экране;
- физический способ взаимодействия с приложением;
- удобство работы вне зависимости от места и времени;
- общее удовлетворение от работы с интерфейсом программы.

Непосредственно разработку интерфейса разделяют на следующие этапы:

1. Создается набросок или прототип интерфейса программы, часто такой прототип передается разработчику из отдела дизайна;
2. Планируется логика перемещения между экранами приложения и методы реализации представления информации на экране;
3. Обсуждается общая концепция и стиль приложения, дизайн приводится к единой и цельной форме;
4. Дизайн реализуется под все допустимые разрешения пользовательских устройств, на всех экранах приложения.

2.4.2 Тестирование мобильных приложений

Для того, чтобы организовать наиболее эффективную работу тестирования мобильных приложений, требуется выполнение предрелизной проверки приложения. Главные этапы проверки показаны в табл. 13

Таблица 13 - Этапы проверки приложения

| Тип проверки | Описание | Этапы |
|-----------------------------------|---|--|
| Контент | Является самым важным элементом любого приложения. Особое внимание уделяется проверке контента. | Проверить все текстовографические блоки; Проверить поддержку языков; Проверить корректное отображение всех элементов; Проверить сопутствующую информацию: дату, число, время и т.д. |
| Отклик в приложении | Если пользователь использует какой-то функционал приложения (нажимаются кнопки, открывается меню, осуществляется поиск), им должен получаться от приложения ожидаемый отклик, который будет свидетельствовать о верной последовательности действий. | Проверить правильное отображение нажатого состояния элементов; Проверить при нажатии отклик кнопок. (выполняют на слабых устройствах); Проверить сообщения приложения при процессах и ошибках (загрузки, ошибки, отсутствие доступа, удаление данных и т.д.); Проверить релевантное и правильное отображение медиа контента (звуки, вибрации). |
| Обновления | Должны быть простые и быстрые для пользователя. Не должны требоваться дополнительные знания. После обновления, приложением не должна теряться накопленная информация о пользователе. Обновления необходимо устанавливать через любое соединение. | Проверить соответствие функционалов старой и новой версии; Проверить адекватность обновления (все ли данные сохранены). |
| Внештатные ситуации (Monkey test) | Приложение должно показывать адекватную реакцию на получение хаотичной информации: попадание без блокировки экрана в карман, нажатие одновременно на несколько пунктов и т. д. | Проверить удобство навигации по приложению; Проверить отсутствие пустых экранов; Проверить одновременное нажатие на все клавиши; Проверить нативные жесты, предусмотренные функционалом. |

| Тип проверки | Описание | Этапы |
|------------------------------------|--|--|
| Мультиплатформенность | Приложение на разных типах устройств должно отображаться правильно. | Проверить работу на retina- экранах элементов интерфейса; Проверить переход приложения от вертикального к горизонтальному положению; Проверить адекватную работу всех медиа файлов на различных операционных системах и устройствах. |
| Операционные возможности устройств | Иногда при одновременной работе нескольких приложений отсутствует возможность открытия нового или корректной загрузки уже открытого. В приложении должны учитываться такие ситуации. | Проверить работу приложения при нехватке памяти; Проверить поведение приложения при открытии в условиях нехватки памяти; Проверить поведение приложения при нехватке свободного места для установки; Проверить поведение приложения при отсутствии на устройстве какой-либо функции (30, 8Б- карта и т. п.); Проверить установку, перенос и удаление приложения. |

2.4.3 Тестирование базы данных

Чаще всего тестирование БД предназначено для решения ряда задач.

А. Тестирование логической модели предназначено, чтобы:

- проверить модель на отсутствие повторяющейся информации и логическую согласованность;
- найти возможности для упрощения логической модели.

Б. Тестирование логической схемы базы данных предназначено, чтобы:

- тестировать на соответствие нормальным формам (обычно 3)
- тестировать на согласованность базы данных (ограничивающие условия, внешние ключи, триггеры)
- тестировать на избыточность данных.

В. Тестирование физической структуры базы данных предназначено, чтобы:

- проанализировать и настроить покрытие индекса;
- проанализировать систему хранения данных (табличные области (Oracle, DB2), группы файлов и массивы данных (MS SQL)), настроить для увеличения надежности и производительности [2,3];

- проанализировать безопасность и разработать предложения для ее улучшения (пользователи, программы и роли пользователя, логины, которые интегрированы с операционной системой, хранимые процедуры)
- проанализировать и реализовать распределение базы данных;
- проанализировать и реализовать стратегии резервного копирования.

Г. Тестирование возможности программирования базы данных предназначено, чтобы:

- проанализировать эффективность хранимых триггеров и процедур;
- оптимизировать запросы, настроить индекс для охвата конкретных запросов;
- проанализировать эффективность клиентской программы.

Существуют отдельные решения для целенаправленного тестирования. К примеру, для того, чтобы проверить некоторые настройки СУБД - тестирование отдельных типов запросов.

Основные недостатки существующих методов тестирования БД с позиции использования ТПП заключаются в отсутствии детального анализа последовательности запросов, которые поступают на вход ИС, невозможности выделения и дальнейшего тестирования отдельных фрагментов БД в соответствии с требованиями ТПП. При сопоставлении требований к модели ТПП с возможностями современных систем тестирования, можно отметить необходимость решения в модели основных задач:

1. Сохранить последовательность запросов, которые поступают в ИС (СУБД) в течение указанного времени наблюдения.
2. Выделить запросы, которые встречаются наиболее часто и (либо) трудоемких запросов по типу выборки и модификации данных.
3. Выделить группы таблиц и (либо) полей, которые связаны с определенным запросом (группой запросов).
4. Выделить группы запросов, которые используют группу таблиц.

5. Многократно «прогнать» тестовую последовательность запросов для фрагмента РБД, чтобы проверить эффективность принятых решений по модернизации БД согласно принятому ТПП.

2.4.4 Генерирование данных для тестирования РБД

Известна множество инструментов генерации тестовых данных (Data Turbo, DTM Data Generator, DBMonster и др.), Которые позволяют заполнить таблицы БД синтаксически корректными данными. При этом испытателю могут быть предоставлены следующие возможности:

- генерация случайных данных заданного формата и в заданном интервале значений;
- генерация данных из списка, с возможностью определения процентного соотношения каждой строки по отношению к другим генерируемых строк;
- генерация данных путем их выборки некоторой таблицы;
- генерация данных путем их выборки из файла;
- генерация инкремента данных, с указанием начального значения и шага;
- генерация данных из существующих библиотек;
- генерация случайных данных по маске;
- генерация данных на основе значений SQL-запросов;
- генерация данных на основе внешних процедур генерации, создаваемых пользователем.

Сопоставления возможностей генераторов данных и требований к модели БД показывает, что не все требования заполнения данными модели БД для ТПП в известных генераторах реализованы. Так, например, границы реальных значений числовых значений и дат не определяются автоматически. Существуют решения о нацеленной генерации данных, но это касается только программ пользователя, а не конкретных таблиц. Известные решения для сохранения конфиденциальности информации, но при генерации данных для конкретной таблицы не соблюдаются реальные соотношения между соседними записями (требования индексации). Отсутствуют процедуры преобразования реальной последовательности запросов в

последовательность запросов к модели БД, отсутствуют процедуры генерирования строк с сохранением реальных длин и соотношений лексикографического порядка, отсутствуют специальные процедуры для замены реальных изображений с БД на правдоподобные для модели.

2.4.5 Прогнозирование развития РБД

Для поддержания хорошей работы ИС не следует ждать момента, когда время выполнения запросов снизится до уровня заметного пользователям системы либо не будет обеспечивать своевременного выполнения их обязанностей. Обычно задачи определения, в какой степени ИС может поддерживать необходимую производительность, решает тестирования производительности или нагрузочное тестирование.

Рассмотрим основные виды нагрузочного тестирования и решаемые ими задачи.

Тестирование производительности (Performance testing) Задачей тестирования производительности является определение масштабируемости приложения под нагрузкой, при этом происходит:

- измерение времени выполнения выбранных операций при определенных интенсивности выполнения этих операций;
- определение количества пользователей, одновременно работающих с программой;
- определение границ приемлемой производительности при увеличении нагрузки (при увеличении интенсивности выполнения этих операций);
- исследование производительности на высоких, пограничных, стрессовых нагрузках.

Стрессовое тестирование (Stress Testing) Стрессовое тестирование позволяет проверить насколько программа пользователя и система в целом работоспособны в условиях стресса и также оценить способность системы к регенерации, то есть к возвращению в нормальное состояние после прекращения действия стресса. Стрессом в данном контексте может быть повышение интенсивности выполнения

операций до очень высоких значений или аварийное изменение конфигурации сервера. Также одной из задач при стрессовом тестировании может быть оценка деградации производительности, таким образом цели стрессового тестирования могут пересекаться с целями тестирования производительности.

Объемное тестирование (Volume Testing) Задачей объемного тестирования является получение оценки производительности при увеличении объемов данных в базе данных программы, при этом происходит:

- измерение времени выполнения выбранных операций при определенных интенсивности выполнения этих операций;
- может проводиться определения количества пользователей, одновременно работающих с программой.

Тестирование стабильности и надежности (Stability/Reliability Testing). Задачей тестирования стабильности (надежности) является проверка работоспособности программы при длительном (многочасовом) тестировании со средним уровнем нагрузки. Время выполнения операций может играть в данном виде тестирования второстепенную роль. При этом на первое место выходит отсутствие «утечек памяти», перезапусков серверов под нагрузкой и другие аспекты, которые влияют именно на стабильность работы.

Анализ способов тестирования производительности показывают, что они не имеют специальных методов и средств для прогнозирования развития ИС на основе анализа временной последовательности запросов, тенденций роста объемов данных в отдельных таблицах. Для адресного применения ПМПП нужна более подробная информация о существующей и прогнозируемой продуктивности ИС.

3 РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТАЛЬНОГО ИССЛЕДОВАНИЯ

3.1 Методика выполнения операций управления мобильностью

При обращении к корпоративным ресурсам извне, возникает ряд проблем администрирования мобильных устройств, работающих с приложением. Шлюз перенаправляет неизвестные запросы к серверу Управления Корпоративной Мобильностью для контроля этих устройств.

После обработки, сервер предоставляет этим устройствам необходимый уровень доступа и уведомляет сервер шлюза о принятых устройствах.

Сотрудникам предприятия предоставлен выбор между корпоративными устройствами и собственными. Это позволяет увеличить эффективность кадрового состава за счет самостоятельного планирования работы без ограничений в пространстве и времени. На рисунке 17 представлен процесс обмена данными между системами и реализации PUSH-уведомлений на устройства. Без подобных уведомлений невозможно своевременное информирование пользователя и обмен контекстными данными в приложении, с которым взаимодействует сотрудник [3].

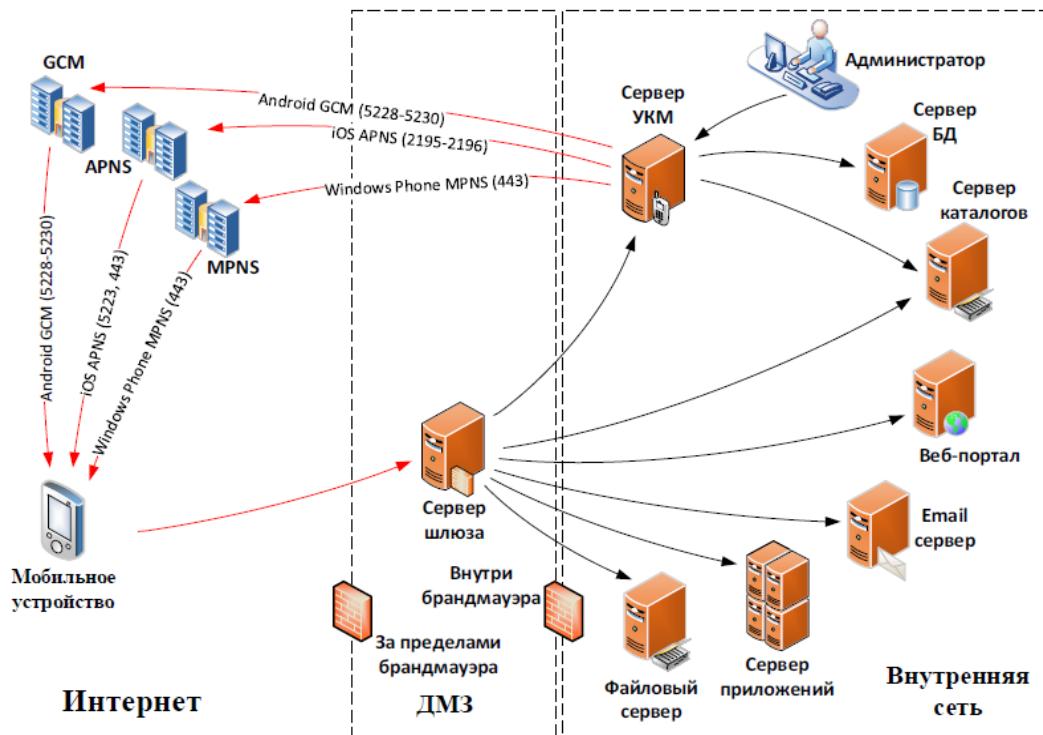


Рисунок 17 - Среда системы Управления Корпоративной Мобильностью (УКМ)

В представленной системе функционируют следующие рабочие процессы:

- регистрация и аутентификация устройства;
- мониторинг пользовательских устройств;
- защита устройств от несанкционированного доступа;
- предоставление устройствам ресурсов компании;
- централизованный контроль над данными, предоставляемыми на устройства.

Непосредственный контроль мобильности происходит в 9 этапов, представленных на рисунке 18.

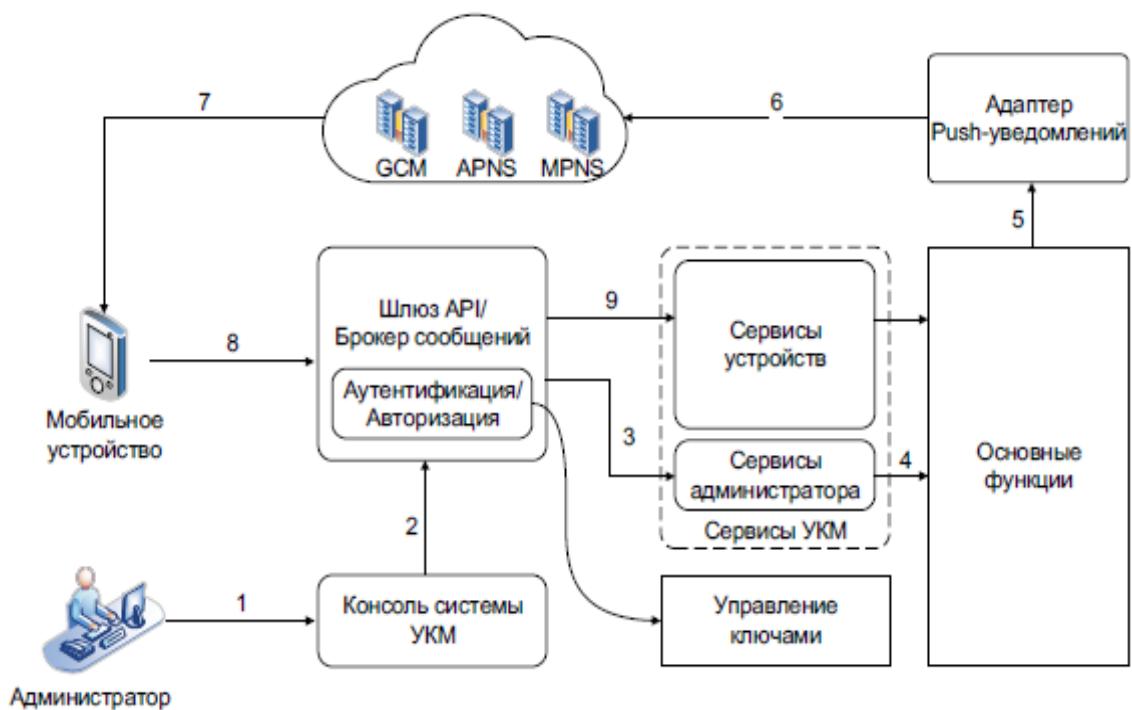


Рисунок 18 – Управление корпоративной мобильностью и реализация PUSH-уведомлений

1. Используя веб-интерфейс, системный администратор задает порядок операций к выполнению на устройстве пользователя;
2. Веб-интерфейс вызывает API-функции на API-шлюзе;
3. Функция запускает backend-сервис;
4. Сервисы запускают выполнение необходимых функций на оборудовании пользователя;

5. После успешного окончания выполнения запрошенных операций вызывается адаптер PUSH-уведомлений;
6. Адаптер запускает необходимую службу (MPNS для Windows Phone, APNS для iOS и GCM для Android);
7. Служба уведомляет пользователя о начале выполнения запрошенной операции;
8. Для получения операции, устройство пользователя вызывает темы на брокере обмена сообщений;
9. Происходит обмен сообщениями между устройством и брокером, которые подписаны и опубликованы по обозначенным темам.

3.2 Математическая постановка задачи

В связи с этим было предложено некоторые общие правила защиты мобильных устройств [2]:

1. Блокирование устройства.

В случае потери мобильного устройства нужно заблокировать устройство через пароль (стойкое или с ограниченное количество попыток ввода), после которых происходит уничтожение данных на устройстве либо блокировка устройства.

2. Пользование криптографическими средствами.

Необходимо пользоваться шифрованием карт памяти, съемных носителей - всего того, к чему может быть получен доступ злоумышленником.

3. Запрещено сохранение паролей в браузере мобильного устройства.

Нельзя сохранять пароли в менеджерах паролей браузеров, даже на мобильных устройствах. Желательно обеспечить установку ограничений на доступ к переписке SMS и почтовой, пользоваться шифрованием.

4. Запрет пользования менеджерами паролей для корпоративных учетных записей.

Существует большое количество приложений, которые на мобильном устройстве созданы для хранения всех паролей. Доступ к программе

обеспечивается через введение мастер-ключа. Если он является недостаточно устойчивым, то вся парольная политика организации может быть скомпрометирована.

5. Запрет установки ПО с непроверенных источников.

Желательно пользоваться ПО известных разработчиков.

6. Пользование корпоративными политиками и средствами антивирусной защиты.

Если данное реально поможет исключить множествоа угроз (включая новые), а при потере или краже устройства, осуществить его блокировку и уничтожение на нем данных.

7. Ограничить список данных, которые можно передавты через облачные сервисы.

Современные мобильные приложения и устройства ориентированы на пользование множеством облачных сервисов. Важно следить, чтобы конфиденциальные данные и данные, которые являются коммерческой тайной, случайно не синхронизировались или не отправлялись в один из таких сервисов.

Для облегчения управления мобильными устройствами работников компаний было сформировано и предложены принципы построения политики защиты мобильных устройств «Bring Your Own Device» и «Mobile Device Management». Они включают в себя вышеуказанные правила защиты мобильных устройств, а также учитывают особенности введения корпоративной деятельности.

MDM, Mobile Device Management, или «администрирование и управление мобильных устройств». Использование решений класса MDM позволяет осуществить управление и контроль над различными типами мобильных устройств.

MDM - это технология управления всеми мобильными устройствами, а технология BYOD - ориентирована на специфику управления устройствами сотрудников в корпоративной среде. BYOD ближе к тактическому и в некоторых аспектах стратегического уровня управления информационными технологиями и информационной безопасностью, тогда как MDM предусматривает прикладную

техническую реализацию, и находится скорее на операционном уровне [5].

В общем случае системы типа MDM - это вспомогательное программное обеспечение, позволяющее управлять мобильными устройствами на каждом этапе жизненного цикла, от инициализации до вывода из эксплуатации [6]. Одна из главных задач MDM - достижение оптимального состояния между безопасностью и удобством использования мобильных устройств, при минимизации затрат на обслуживание. Это достигается за счет таких возможностей систем типа MDM [3]:

- централизованное управление мобильными настройками (парольные политики, параметры шифрования)
- запрет запуска нежелательных приложений;
- инвентаризация программных и аппаратных средств на мобильных устройствах;
- устанавливать политики безопасности, сертификаты безопасности и пароли на устройствах пользователей;
- настраивать WiFi и VPN соответствии с корпоративными стандартами;
- устанавливать мобильные приложения, вести черный/белый список программ;
- предоставлять удаленную поддержку пользователей;
- удаленно блокировать устройство и уничтожать корпоративные данные в случае его потери или кражи;
- настраивать ограничения для устройств, например, запрет передачи данных в роуминге, отключение камеры, использование USB, Bluetooth, магазинов приложений (AppStore, Google Play и др.)
- контролировать соответствие устройства установленным корпоративным политикам;
- выполнять резервное копирование и шифрование данных на устройствах;
- осуществлять групповую подготовку, конфигурирование и обслуживание устройств.

Исходя из функциональных возможностей систем типа MDM можно предложить общий сценарий внедрения MDM:

1. Инициализация, в которую входят такие этапы - инвентаризация, анализ потребностей пользователей в части использования устройств и информационных ресурсов, классификация информационных ресурсов, определение информационных рисков и моделирование угроз.

2. Разработка. Разработка политики и стандарта применения мобильных устройств и разработка регламентов защиты и управления мобильными устройствами.

3. Внедрение. Настройка базовых компонентов управления, опытная эксплуатация механизмов управления, масштабирования системы.

4. Сопровождение. Реализация сервиса сопровождения мобильных устройств, контроль использования мобильных приложений, контроль применения и соблюдения корпоративных политик для мобильных устройств, мониторинг деятельности пользователей.

5. Вывод устройств из обращения. Разработка и выполнение правил утилизации мобильного устройства, реализация регламента действий в случае кражи или потери мобильного устройства.

С другой стороны, решением вопроса обеспечения политики безопасности информации в корпоративной сети является внедрение технологии BYOD. Системы типа BYOD - это программное обеспечение, которое управляет доступом к сети пользователей, подключающихся к различным сетевым устройствам (точки Wi-Fi, коммутаторы, VPN концентраторы). И в зависимости от того, находится ли пользователь в интернет или внутри сети, это программное обеспечение устанавливает соответствующий профиль с заданными правами [4]. А интеллектуальная сетевая инфраструктура выполняет задачи по реализации прав доступа для каждого конкретного пользователя. Основой же для применения такого современного подхода является разработка корпоративных политик использования личных устройств [4]. Для корректного обеспечения такого подхода, системы типа BYOD имеют такие возможности [1]:

- плавная размещения с использованием автоматической регистрации и создания профиля устройства обеспечивает сохранение уровня

производительности пользователя;

- конвергентное управления для проводных и беспроводных сетей, клиентских устройств;
- оптимизированное планирование ресурсов с полным отслеживанием трафика и действий пользователей при использовании концепции BYOD;
- полная масштабируемая производительность проводных и беспроводных сетей;
- модульная система прямого управления позволяет добавлять ресурсы и функции по мере необходимости.

Имея такие функциональные возможности, технология BYOD имеет такой сценарий внедрения в корпоративную сеть:

1. Планирование (проектирование). Определение стратегии BYOD, разработка корпоративной политики, инвентаризация, анализ потребностей пользователей в части использования устройств и информационных ресурсов, моделирование угроз информационной безопасности, разработка регламента управления рисками, разработка регламента реагирования на инциденты безопасности, разработка нормативных документов, разграничитывающие сферы ответственности сотрудников и организации при использовании персональных устройств, разработка стандарта по применению личных устройств в корпоративной среде.

2. Внедрение. Внедрение централизованных механизмов управления персональными устройствами сотрудников, интеграция централизованной системы управления действующим инфраструктурными и прикладными системами и средствами защиты, обучение персонала.

3. Контроль (периодический аудит).

4. Сопровождение, модернизация - работа службы технического сопровождения.

В соответствии с этим можно выделить ключевые преимущества систем типа BYOD:

- идентификация, управление сетью и доступом к приложениям для

любых пользовательских устройств;

- безопасность доступа к сети и приложений независимо от местонахождения;
- организация проводных и беспроводных сетей с использованием единого интерфейса прямого управления;
- упрощение проектирования сетей для обеспечения масштабируемости проводных и беспроводных локальных сетей;
- обеспечение доступа для мобильных устройств к мультимедиа.

После определения величины затрат ($C_{\text{опт}}$) на обеспечение защиты объекта необходимо сформировать систему элементов объекта, удовлетворяющую заданным ограничениям.

Количество функций определяет количество возможных вариантов проникновения угрозы на объект с мобильной корпоративной информационной системой. Угрозу можно остановить через обрыв маршрута как минимум в одной дуге. Существует необходимость в исключении всех ста маршрутов проникновения. Другими словами, перед нами задача о нахождении минимального сечения на графе, решить которую можно способом определения минимального покрытия на матрице инцидентности.

В задаче о покрытии необходимо прийти к следующей цели: минимально возможным количеством дуг перекрыть все предполагаемые маршруты проникновения. Результатом будет являться минимальный набор дуг, исключающих проникновение, т.е. реализацию угрозы.

Дуга может представлять собой тип варианта защиты, характеризуемый стоимостью и вероятностью защиты. Решением задачи о покрытии будет максимум эффективности, поэтому в качестве основного показателя остановимся на количестве перекрываемых маршрутов, минимальной избыточности. В итоге, результат – это множество покрытий, в котором каждый возможный вариант перекрывает маршруты проникновения. Математическая, задача звучит следующим образом: необходимо минимизировать целевую функцию

$$L = \sum_{j=1}^n \tilde{n}_j x_j \rightarrow \min \quad (1)$$

при ограничениях

$$\begin{aligned} \sum_{j=1}^n a_{ij} x_j &\geq 1, \quad i = \overline{1, m}; \\ x_j &\in \{0, 1\}; \quad c_j \geq 0; \end{aligned} \quad (2)$$

где

$$x_i = \begin{cases} 1, & \text{если } i \text{ технологическая операция выполняется,} \\ 0, & \text{в противном случае.} \end{cases} \quad (3)$$

Ограничение (2) означает, что каждый маршрут проникновения должен покрываться (прерываться). Исходные данные задаются с помощью матрицы инцидентности $A = \|\alpha_{ij}\|$.

где $i = 1, \dots, m$ - количество маршрутов, которые должны быть перекрыты;

$j = 1, \dots, n$ - количество действующих элементов (дуг), которые могут перекрывать (покрывать) определенные маршруты;

$$a_{ij} = \begin{cases} 1, & \text{если } i \text{ маршрут покрывается } j \text{ действующим элементом;} \\ 0, & \text{не выполняется.} \end{cases} \quad (4)$$

Для решения задачи о покрытии предлагается использовать ранговый подход.

3.3 Разработка приближенного и точного алгоритмов решения задачи о покрытии на основе рангового подхода и проведение экспериментального исследования

Пусть A^T – транспонированная матрица смежности графа G с единичными диагональными элементами. Задача нахождения наименьшего доминирующего множества графа G равносильна задаче нахождения такого наименьшего множества столбцов в матрице A^T , что каждая строка матрицы будет содержать единицу как минимум в одном из исследуемых столбцов. Эта задача представляет

собой поиск наименьшего множества столбцов, «покрывающих» единицами все строки, получила названия задачи о минимальном покрытии (ЗНП). Матрица, состоящая из 0 и 1, не всегда будет представлена квадратной. Каждому столбцу j (вершине x_j) в матрице A^T сопоставляется некоторый вес, и требуется найти покрытие с наименьшей общей стоимостью. В случае же равенства коэффициентов задача трансформируется в задачу минимизации числа столбцов, покрывающих все строки в матрице A^T .

Рассмотрение алгоритмов решения ЗНП начнем с приближенного алгоритма решения задачи (1 – 3) для случая равенства коэффициентов c_i с использованием правил объединения L_1 и L_2 . Наиболее простым и естественным правилом отсечения путей L_1 во множествах $m_{si}^r, i = \overline{r, n}$ является удаление путей $\mu_{si}^r \in m_{si}^r$, у которых $d_i^n = \infty$, поскольку эти пути не удовлетворяют свойству v . Тогда ЗНП можно рассматривать как задачу определения кратчайших путей на основе рангового подхода [8-9] в графе D_Δ , с учетом принципа оптимизации по направлению. Пошаговое описание алгоритма определения кратчайших путей в графе D_Δ , удовлетворяющих свойству v , имеет вид:

Приближенный алгоритм решения невзвешенной ЗНП (процедура A_k)

Шаг 1. Формируем калибровочные вектора $H^{(i)}$, выделяем множества путей $\{\mu_{si}^{r=1}\}$, удовлетворяющие свойству v . Если все эти множества пусты, то алгоритм заканчивает работу, т.к. это означает, что покрытия не существует. Если же есть хотя бы одно непустое множество, то определяем в каждом из них пути $\mu_{si}^{*, r=1}$, минимальные по весу d_i^n и делаем переход к шагу 3.

Шаг 2. На основе пути множества путей $\{\mu_{si}^{*, r}\}$ текущего ранга формируем множества путей μ_{si}^{r+1} следующего ранга, удовлетворяющих свойству v . Выделяем в них пути $\mu_{si}^{*, r+1}$ с наименьшим значением d_i^n (правило L_2) и переходим к выполнению следующего шага.

Шаг 3. Если есть $d_i^n = 0$, то алгоритм заканчивает работу, иначе – выполняем шаг 4.

Шаг 4. Если есть $d_i^{\pi} = 1$, то формируем путь следующего ранга с $d_i^{\pi} = 0$ и алгоритм заканчивает работу, иначе – значение r увеличиваем на 1 и переходим к шагу 2. Приведенный алгоритм отображает основное функциональное уравнение динамического программирования, которое для данной задачи можно представить в следующем виде

$$d_{\min}^{r=k}(i) = \min_j \left[d_{\min}^{r=k-1}(s, j) + d^{r=k}(j, i) \right], \quad (5)$$

где $d_{\min}^{r=k-1}(s, j)$ – условное минимальное расстояние от вершины s до вершины j в D_Δ , найденное на $(k-1)$ -м шаге работы процедуры;
 $d^{r=k}(j, i)$ – величины весов вершин, образующих пути к вершинам $i=(1, n)$ на k -м шаге.

Уравнение (5) позволяет по цепочке находить условные оптимальные решения. Таким образом, определение калибровочных векторов $H^{(i)}$ эквивалентно маркировке путей в графе D_Δ от конца к началу, а использование затем процедуры A_k есть ни что иное, как идентификация кратчайшего пути по сделанной маркировке на основе функционального уравнения динамического программирования (5), когда процесс программирования разворачивается от начала к концу. В случае взвешенных столбцов в матрице B пошаговое описание приближенного алгоритма A_k^B решения задачи (1 – 3) с использованием правила отсечения L_2 остается тем же, но при этом поиск кратчайших путей идет по весовой характеристике d_i^{ne} , определяемой соотношениями (5).

Приближенный алгоритм решения взвешенной ЗНП (процедура A_k^B)

Шаг 1. Формируем калибровочные векторы $H^{(i)}$, выделяем множества путей $\{\mu_{si}^{r=1}\}$, удовлетворяющие свойству v . Если все эти множества пусты, то алгоритм заканчивает работу, т.к. это означает, что покрытия не существует. Если же есть хотя бы одно непустое множество, то определяем в каждом из них пути $\mu_{si}^{*, r=1}$, минимальные по весу $d_i^{\pi_B}$ и переходим к шагу 2.

Шаг 2. Если есть пути с $d_i^{\text{пп}} = 0$, то выделяем среди них путь, минимальный по весу, определяющий локальный экстремум на ранге, и переходим к следующему шагу.

Шаг 3. На основе пути множества путей $\{\mu_{si}^{*r}\}$ текущего ранга r формируем множества путей μ_{si}^{r+1} следующего ранга в соответствии с правилом L_2 и переходим к выполнению следующего шага.

Шаг 4. Если все множества следующего ранга пусты, то выделяем из множества локальных экстремумов глобальный и алгоритм заканчивает работу. Если же есть хотя бы одно непустое множество, то увеличиваем значение текущего ранга $r:=r+1$ и переходим к шагу 2.

Точное решение задачи (1-3) в случае равенства коэффициентов c_i возможно только с использованием правил L_1 и р. При этом пошаговое описание алгоритма, реализуемого на основе процедуры A_0 , имеет вид:

Точный алгоритм решения невзвешенной ЗНП (процедура A)

Шаг 1. Формируем калибровочные вектора $H^{(i)}$, выделяем множества путей $\{\mu_{si}^{r=1}\}$, удовлетворяющие свойству v . Если все эти множества пусты, то алгоритм заканчивает работу, т.к. это означает, что покрытия не существует. Если же есть хотя бы одно непустое множество, то для всех вершин графа D_Δ вычисляем вспомогательные вектора z_i , определяем для каждого пути характеристики $d_i^{\text{п}}$ и d_i^o в соответствии с соотношениями (6 – 9) и переходим к выполнению следующего шага.

Назовем наименьшее число вершин d_i^o , подсоединение которых к пути μ_{si}^r удовлетворяет неравенству

$$\sum_{j=1}^{d_i^o} \alpha_{i+j} \geq a_i^H, \quad (6)$$

оптимистическим негарантированным прогнозом, определяющим минимальное число вершин d_i^o , которое необходимое присоединить к пути μ_{si}^r для того, чтобы получить путь μ_{st}^{r*} , соответствующий покрытию в матрице В. Вершины i

отсортированы в графе D_Δ в порядке возрастания γ_i и, следовательно, в порядке убывания α_i . Введем для каждой вершины i вспомогательный вектор $z_i(z_{i+1}, z_{i+2}, \dots, z_{i+k})$, где $i + k \leq n$, а компоненты вектора определяются следующими рекуррентными соотношениями

$$\begin{aligned} z_{i+1} &= \alpha_{i+1} \\ z_{i+2} &= z_{i+1} + \alpha_{i+2} \\ &\dots \\ z_{i+k} &= z_{i+(k-1)} + \alpha_{i+k} \end{aligned} \tag{7}$$

Вспомогательный вектор z_i для вершины i графа D_Δ позволяет определить оптимистический негарантированный прогноз для произвольного пути μ_{si}^r , характеризуемого вектором $H^{(i)}$. Для этого определяется величина $a_i^H = m - p_i$, и тогда

$$d_i^o = j, \tag{8}$$

где j – значение, начиная с которого выполняется неравенство

$$z_{i+j} \geq a_i^H; j = \overline{1, k}; (i + j \leq n). \tag{9}$$

Если величина пессимистической оценки пути μ_{si}^r $d_i^\pi = \infty$, то и $d_i^o = \infty$.

Шаг 2. На основе пути множества путей $\{\mu_{si}^{*,r}\}$ текущего ранга r формируем множества путей μ_{si}^{r+1} следующего ранга $r+1$ на основе рекуррентного соотношения (10)

$$m_{sp}^{r+1} = \left\{ \forall \left(\mu_{si}^r \in m_{si}^r \right) \cup (i, p) \right\}. \tag{10}$$

определяем характеристики этих путей d_i^π и d_i^o в соответствии с (6 – 9) и выделяем коридор во множестве m_{si}^{r+1} по правилу p , определяемому соотношением (11):

$$\begin{aligned} \forall \left(\mu_{si}^r \in m_{si}^r \right) : \left[\mu_{sp}^{r+1} = \min_{d_i^r} \left\{ \mu_{si}^r \cup (i, p) \right\} \right], \\ p = \overline{r+1, n}; i = \overline{r, n}, \end{aligned} \tag{11}$$

Шаг 3. Если есть $d_i^\pi = 0$, то алгоритм заканчивает работу, т.к. именно этот путь образует минимальное покрытие, иначе - выполняем шаг 4.

Шаг 4. Если есть $d_i^n = 1$, то формируем путь следующего ранга с $d_i^n = 0$ и алгоритм заканчивает работу, иначе – значение r увеличиваем на 1 и переходим к шагу 2.

При исследовании ЗНП в качестве эталонного был взят алгоритм метода ветвей и границ (МВГ), использующий стратегию поиска в глубину. Выбор этого метода связан с тем, что он универсален, прост в реализации, не требователен к машинным ресурсам и считается наиболее быстродействующим из существующих. Что же касается стратегии поиска, то она была выбрана по следующим соображениям. Считается, что использование стратегии поиска в глубину позволяет решать задачу с меньшим расходом ресурсов ЭВМ, но время поиска при этом увеличивается по сравнению с поиском в ширину. И на малых размерностях это действительно так. Но с увеличением размерности задачи объем промежуточных данных возрастает на столько, что оперативной памяти (ОЗУ) для их хранения оказывается недостаточно. Поэтому эти данные приходится хранить на внешних запоминающих устройствах (ВЗУ), что существенно увеличивает время решения задачи, так как операция чтения и записи на ВЗУ существенно больше, чем у ОЗУ. Кроме того, различные модификации МВГ, определяющие до начала работы основного алгоритма верхнюю оценку решения, позволили практически выровнять количество анализируемых вариантов для обеих стратегий. Поскольку же нас в первую очередь интересовали задачи большой размерности, то в качестве эталонного был выбран МВГ со стратегией поиска в глубину. Использование МВГ позволяет уменьшить количество анализируемых вариантов, но проблема состоит в том, что, даже получив решение, алгоритм продолжает проверять те ветви дерева путей, вес (ранг) которых меньше полученного решения. Таким образом, даже если предположить, что верхняя оценка решения совпала с оптимальным значением R^* , количество векторов (узлов ветвления), построенных

МВГ, будет равно сумме R^* элементов ряда треугольника Паскаля $\sum_{r=1}^{R^*} C_n^r$.

Алгоритмы же, основанные на ранговом подходе, за счет предварительной калибровки и прогнозирования конечного результата позволяют на каждом ранге,

начиная с первого, отсекать те пути, которые алгоритм определяет, как неперспективные. Под неперспективными понимаются пути, которые в принципе не могут дать покрытие, а также те, которые дадут покрытие на ранге, превышающем прогнозируемое оптимальное решение. Другими словами, в отличие от МВГ, отсекаются не только те ветки, длина которых больше оптимального значения, но и те, которые не содержат решения в принципе.

3.4 Подготовка и оформление доказательной базы диссертационного исследования согласно рабочей гипотезе

Оценка погрешностей алгоритмов решения задач дискретной оптимизации представляет интерес с точки зрения возможности их сравнения. Представим описание понятию “оптимационная задача” – комбинаторная оптимационная задача $\Pi(n)$ размерности n -есть задача минимизации или максимизации определенной целевой функции, состоящая из трех частей:

1. Множества $Z_n(n)$ индивидуальных задач размерности n .
2. Множества допустимых решений $\Omega_n(I(n))$ для каждой индивидуальной задачи $I(n) \in Z_n(n)$.
3. Функции f_n , сопоставляющей каждой индивидуальной задаче $I(n) \in Z_n(n)$ и каждому допустимому решению $\sigma_n(n) \in \Omega_n(I(n))$ число $f_n(I(n), \sigma_n(n))$, называемое величиной решения.

Размерность задачи $I(n)$ и соответственно функция размера входа алгоритма А, решающего задачу I, представляется в виде определенной последовательности знаков, необходимых для корректного кодирования, размер n входа при этом определяется, как число символов в этой последовательности. Если $\Pi(n)$ задача максимизации, то оптимальным решением индивидуальной задачи $I(n) \in Z_n(n)$ является такое допустимое решение $\sigma_n^*(n) \in \Omega_n(I(n))$, что для всех $\sigma_n(n) \in \Omega_n(\Pi(n))$ выполняется неравенство $f_n(I(n), \sigma_n^*(n)) \geq f_n(I(n), \sigma_n(n))$. Для обозначения величины $f_n(I(n), \sigma_n^*(n))$ оптимального решения задачи $I(n)$ используем $OPT(I(n))$. Анализ погрешности проводим на примере задачи (0,1)-рюкзак, в связи с этим индекс в

далее может опускаться. Через $A(I(n))$ обозначим величину $f_n(I(n), \sigma_n(n))$ того возможного $\sigma_n(n)$, которое алгоритм A строит по $I(n)$. Если $A(I(n)) = OPT(I(n))$ для всех $I(n) \in Z_n(n)$, алгоритм предоставит решение задачи Π .

В частном порядке разберем наиболее естественную оценку отклонения от оптимума, а именно для всех индивидуальных задач $I(n)$ имеет место оценка

$$|A(I(n)) - OPT(I(n))| \leq k, \quad (12)$$

где k – константа.

Если приближенный алгоритм A решает задачу $(0,1)$ -рюкзак на основе выделения коридора по величине функционала при использовании рангового подхода, то

$$\lim \Delta = 0,$$

$$\text{где } \Delta = |A(I(n)) - OPT(I(n))|.$$

Условно при экспериментальном анализе погрешностей вычислений определяются три зоны – зоны 1 и 3 соответствуют решениям, согласно условию которых алгоритмы дают точные решения. Требуется доказать ее справедливость для зоны 2, лежащей на интервале [от $r_{cp} = n/3$ до $r_{cp} = 2n/3$], в которой при выделении коридора наблюдается наибольший рост допустимых решений в самом коридоре. В соответствии с треугольником Паскаля, предельное значение числа различных допустимых решений, попадающих во вторую зону, не может превысить $f(r = n/2, n) = C_n^{r=n/2}$, с учетом соотношения Стирлинга

$$\lim_{n \rightarrow \infty} \frac{n!}{n^n e^{-n\sqrt{n}}} = \sqrt{2\pi}$$

имеем

$$C_n^r = f(r, n) = \frac{1}{\sqrt{2\pi n}} \cdot \frac{\left(\frac{n}{r}\right)^{n+0,5}}{\left(\frac{n}{r} - 1\right)^{n-r}}. \quad (13)$$

При $r = n/2$ соотношение (13) примет вид

$$f(r = \frac{n}{2}, n) = \frac{2^n}{\sqrt{\pi n}} = \Psi(n). \quad (14)$$

Вводим функцию плотности числа единиц функционала, приходящихся на одно решение, в выделенном на ярусе r коридоре Δf_r

$$\Lambda = \frac{\Delta f_r}{\Psi(n)} = \frac{\Delta f_r \sqrt{2\pi n}}{2^n}. \quad (15)$$

При переходе от яруса r к ярусу $r = r + 1$ ширина коридора Δf_r не возрастает $\Delta f_{r+1} \leq \Delta f_r$. При $n \rightarrow \infty$ различные решения $A(I(n))$ в коридоре будут отличаться друг от друга на величину, не превышающую Λ . Следовательно, $\Delta \leq \Lambda$. Отсюда следует, что

$$\lim_{n \rightarrow \infty} \Delta \leq \lim_{n \rightarrow \infty} \frac{\Delta f_r \sqrt{2\pi n}}{2^n} = 0,$$

что и требовалось доказать. При фиксированном значении $n = n_0$, $\Lambda = \Lambda_0$ соответственно выполняется неравенство $\Delta \leq \Lambda_0 = \text{const}$.

Поскольку принцип выделения коридора такой же и в ЗНП, то данная теорема справедлива и в случае применения рангового подхода к решению ЗНП. Таким образом при использовании принципа выделения коридора для решения задач ЦЛП с БП увеличение размерности задачи приводит к уменьшению погрешности в среднем, что является важным достоинством рангового подхода к решению задач ЦЛП с БП по сравнению с известными.

3.5 Обработка результатов экспериментальных исследований в соответствии с рабочей гипотезой

На рис. 18 показана зависимость количества обработанных векторов от ранга решения задачи размерности 20×200 для описанных алгоритмов. Для наглядности на этом же графике приведена кривая "треугольника Паскаля". Кривые на графиках подтверждают, что сложность рассмотренных приближенных алгоритмов не может превосходить $O(n^3m)$, поскольку всего ярусов n , на ярусе число формируемых векторов не превосходит n^2 , и число элементов, анализируемых в

каждом векторе, не превосходит m . Для точных алгоритмов полученная оценка временной сложности в среднем также соответствует $O(n^3m)$. Что же касается МВГ, то с увеличением ранга решения МВГ приближается к полнопереборному алгоритму.

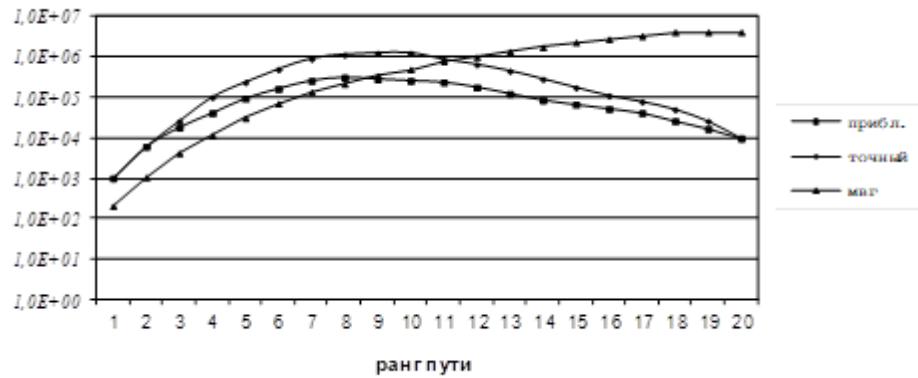


Рисунок 18 - Зависимость количества элементарных операций от ранга пути

Справедливо ради нужно отметить, что количество элементарных операций, необходимых для построения одного вектора, в МВГ меньше, чем в алгоритмах, основанных на ранговом подходе, это компенсируется уменьшением числа обрабатываемых векторов. Графики зависимостей числа элементарных операций и времени решения от ранга пути представлены на рисунках 20 – 24.

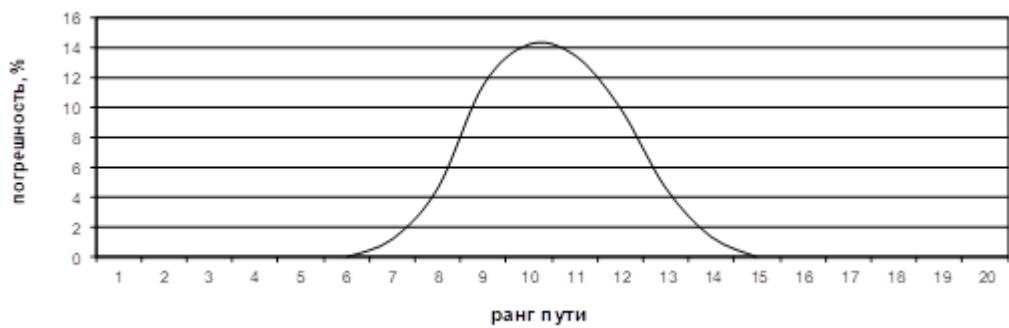


Рисунок 20 - Зависимость количества элементарных операций от ранга пути

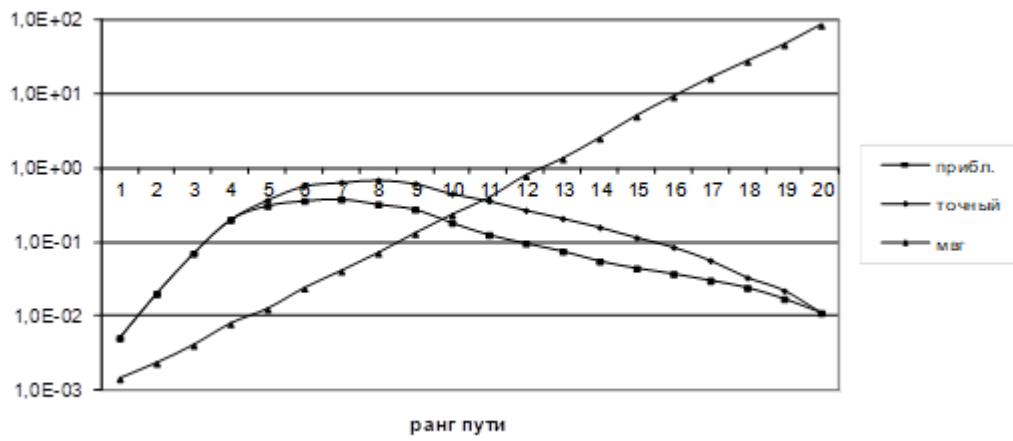


Рисунок 21 - Зависимость времени решения от ранга пути

Что же касается зависимости погрешности приближенного алгоритма от ранга пути, то она возникает при попадании решения во вторую условно выделенную зону и колеблется в пределах от 0 до 15% (рис. 22), причем количество неточных решений не превосходит 10% от общего числа опытов.

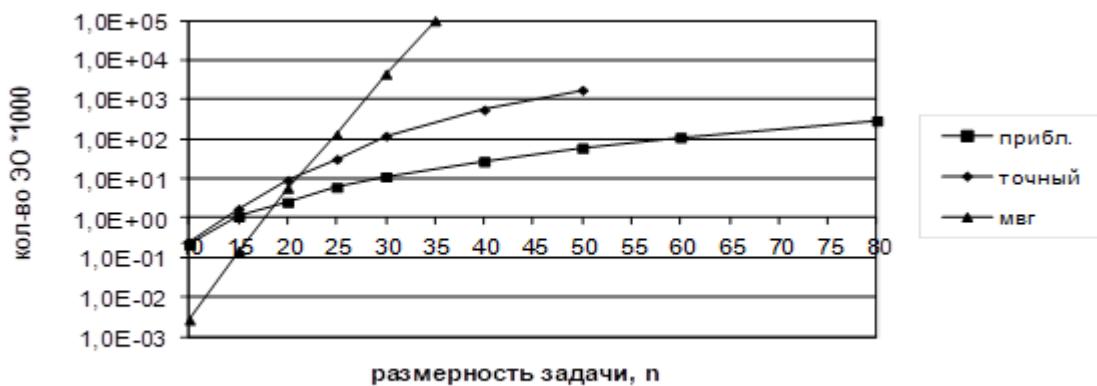


Рисунок 22 - Зависимость погрешности от ранга пути

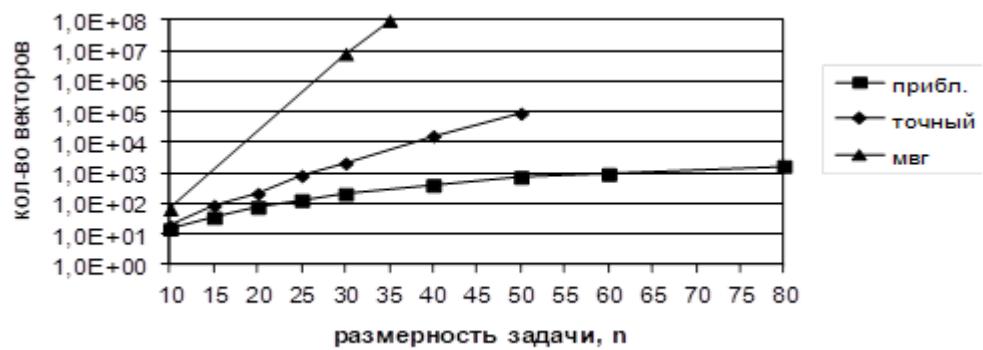


Рисунок 23 - Зависимость количества ЭО от размерности задачи

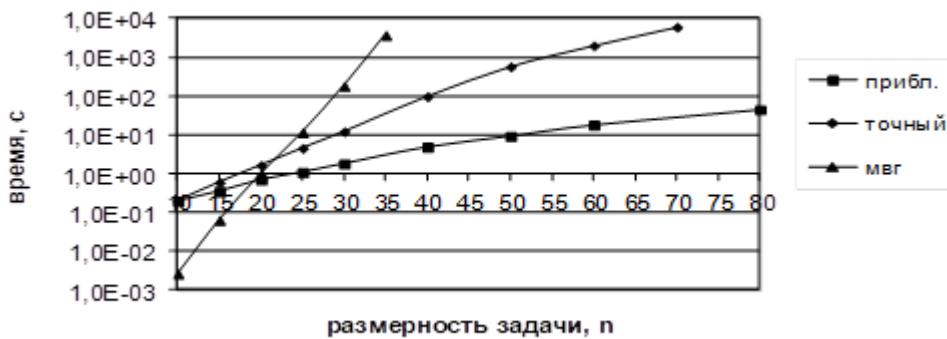


Рисунок 24 - Зависимость времени решения от размерности задачи

Для проведения экспериментального исследования и сравнения метода на основе рангового подхода разработано программное средство, интерфейс представлен на рис. 25-31. Листинг программы реализации методов для сравнения показан в приложении В.

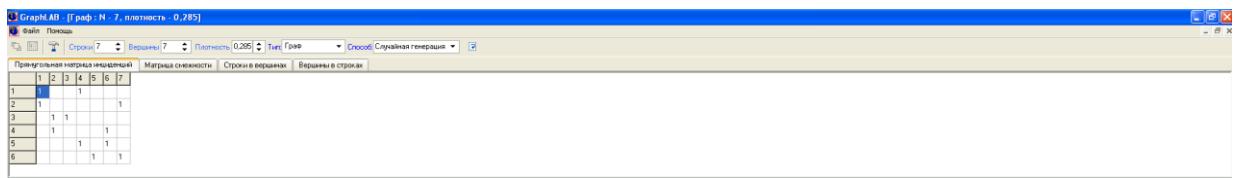


Рисунок 25 - Прямоугольная матрица инциденций графа

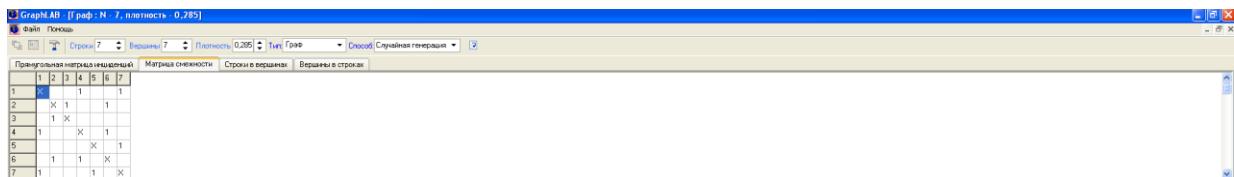


Рисунок 26 - Матрица смежности

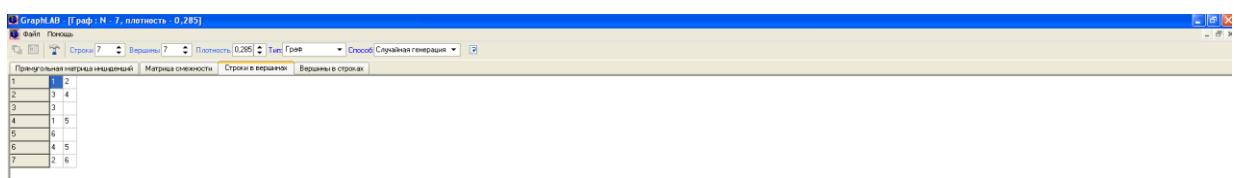


Рисунок 27 - Строки в вершинах



Рисунок 28 - Вершины в строках

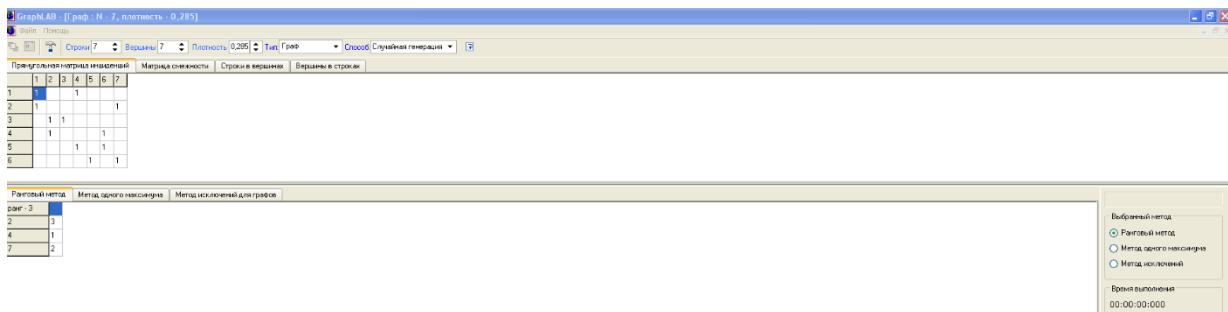


Рисунок 29 - Результат решения задачи на основе рангового подхода

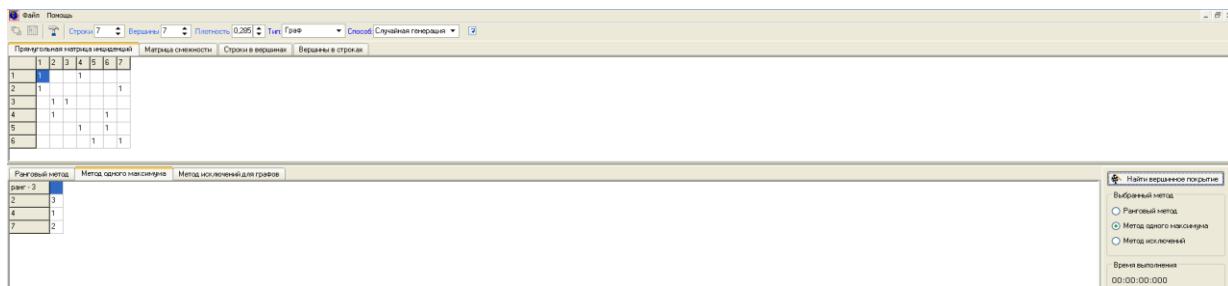


Рисунок 30 - Результат решения задачи методом основного максимума

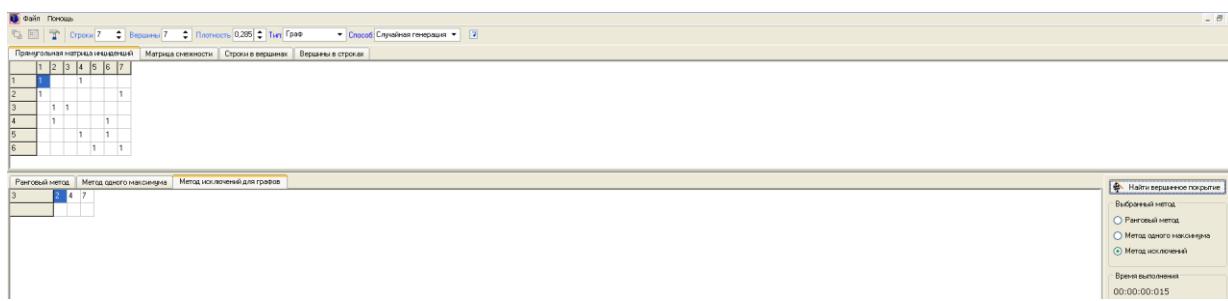


Рисунок 31 - Результат решения задачи методом исключения для графов

ЗАКЛЮЧЕНИЕ

По результатам проделанной работы можно сделать вывод, без внедрения информационных систем на платформе 1С, невозможно привести в полный порядок всю документацию по всему холдингу. Ряд документов по какому-либо проекту не предоставлялся бы специалисту в своевременной форме, страдала бы сопоставимость и достоверность данных. В интересах руководства по первому требованию составлять отчеты по деятельности холдинга во время и в качественной форме. 1С: «Консолидация холдинга» предоставляет доступ специалисту ко всем всем документам в упрощенной форме.

Оценка полноты решений, принятых при внедрении нескольких информационных систем на базе 1С, привели к порядку в холдинге, и добавили новые возможности в документах:

- Организация своевременного ввода первичной информации на местах возникновения;
- Организация сопредельного учета затрат и расчета себестоимости, в т.ч. учет затрат при производстве продукции из давальческого сырья;
- Организация бухгалтерского и налогового учета и обязательной отчетности;
- Организация бухгалтерского и налогового учета в соответствии с требованиями ОАО «Татнефть» для целей своевременной отчетности в рамках КГН и консолидированной отчетности;
- Создание нормативной базы для обслуживания оборудования (более 150 тыс. единиц);
- Ведение технической документации на оборудование;
- Планирование обслуживания активов предприятия и ресурсов для его проведения, в том числе: составление и согласование ключевыми ответственными лицами графиков планово-предупредительных работ;
- Управление ремонтами в соответствии с документами: - Общезаводская инструкция о порядке безопасного проведения ремонтных работ на объектах ОАО «ТАНЕКО»;

- Общезаводское положение о системе планово – предупредительного ремонта и межремонтном обслуживании технологического оборудования в ОАО «ТАНЕКО»

Разработка рекомендаций. Учитывая положительный опыт построения информационных систем на платформе 1С как на уровне холдинга, так и на дочерних предприятиях принято решение о создании Корпоративной Информационной системы 1С. [12, с. 40]

Один из современных трендов – централизация обслуживающих функций и заинтересованность управляющих компаний в повышении контроля не только за финансовыми потоками, но и операционной деятельностью бизнес единиц. С точки зрения бюджетирования смещение акцента от бюджетирования к лимитированию и контролю расходов.

Результаты оценки технико-экономической эффективности внедрения:

- Ведение регламентированного кадрового учета, в соответствии со стандартами ОАО «ТАНЕКО»;
- Формирование отчетности по движению, численности и статистической отчетности по персоналу, согласно утвержденным формам в ОАО «ТАНЕКО», а также отчетности по фонду заработной платы в различных разрезах;
- Автоматизация расчета заработной платы, учета и инвентаризации резервов на отпуска и годовое вознаграждение;
- Автоматизация блока по «Охране труда» (учет вредных условий, учет времени работы сотрудников во вредных условиях, направления на медицинские осмотры);
- Средства для управления процессами обучения и аттестации персонала.
- Автоматизация полного жизненного цикла договора (создание, согласование, исполнение);
- Хранение истории согласования редакций договора;
- План-фактный анализ освоения и финансирования этапов договора;
- Оперативный контроль процесса исполнения договора.

Результаты оценки научно-технического уровня. Необходимость соблюдения государственных стандартов по ведению документации в ходе деятельности холдинга, заставляет шагать в ногу со временем. Информационная система на платформе 1С, позволит контролировать документооборот в наглядном виде, что позволит избежать дальнейших проблем. Каждому документу будет присвоена своя адресная ячейка.

В составлении отчетов, необходимо придерживаться нескольких рекомендаций на случай ликвидации возникших проблем. При финансовом анализе – вся документация холдинга должна быть как на ладони. В такой ситуации работа экономистов будет более продуктивной. Базовая оценка деятельности холдинга, предполагает изучение бухгалтерской отчетности за определенный период времени. Данный анализ может быть использован в фундаментальной исследовательской работе, в которой тщательно анализируют финансовые показатели холдинга, что позволяет выделить отрасли в кампании для дальнейшего в них инвестирования. Такой подход позволяет вести успешную деятельность во всем холдинге «Татнефть».

Информационные системы на платформе 1С: ежегодно обновляются, добавляют дополнительные функции в программу, с целью облегчения работы любому специалисту. Без наличия информационных программ, приводящих все документы в порядок, в непосредственных частях организации творился бы полный хаос.

Исследование автоматизации бухгалтерского учета на промышленных предприятиях выявило, что 60% используют программу автоматизации учета "1С: Предприятие 7.7", что обусловило предложение создания аналитической подсистемы данной программе, которая обеспечит автоматический перенос данных учета для проведения анализа и небольшие потери средств. Подсистема должна учитывать организационные и методические аспекты внутрихозяйственного экономического анализа.

Построение функциональной структуры подсистемы определяют задачи, поставленные при анализе, их взаимосвязи и последовательность решения.

Модуль "Управление аналитическим процессом" дает возможность эффективного планирования аналитического процесса, обеспечение хранения и составления документов работы аналитиков, пояснительных записок и справок, контроль подготовки и проверки информации для анализа с помощью набора требований к учетной информации и автоматизированных приемов контроля, управления системой защиты информационной базы.

Модуль "Документация" предусматривает формирование базы данных, содержащей формы документов и примеры их заполнения по всем направлениям документирования аналитической деятельности. Обеспечение деятельности аналитиков в части нормативных документов происходит с помощью создания модуля "Нормативная база".

Модули анализа ресурсов и их использования формирует набор показателей анализа рынка, ценовой политики, системы распределения продукции, конкурентоспособности стадий жизненного цикла продукции; анализа состава ресурсов предприятия, их изменения по сравнению с предыдущими периодами, эффективности использования, влияния факторов на изменение показателей с целью определения резервов роста эффективности их использования; анализа эффективности инвестиций, инвестиционной привлекательности, окупаемости инвестиционного проекта; объемов роста производства и реализации продукции, измерения резервов снижения себестоимости продукции.

Модули набора финансовых коэффициентов для оценки финансового состояния, направлений и показателей для анализа структуры финансовых результатов от различных видов деятельности, их изменения по сравнению с предыдущими периодами, порядка формирования, влияния факторов на изменение прибыли с целью определения резервов его роста, анализ рентабельности; финансовых показателей для оценки текущей неплатежеспособности и вероятности банкротства предприятия.

Модуль «Прогнозный анализ» обобщает выявленные резервы по всем видам ресурсов.

Успешная реализация аналитических модулей требует интеграции программ направлений: бухгалтерской, правовой, экономического анализа и управления.

Представление n -мерного единичного куба в виде графа D_Δ в сочетании с разработанными правилами отсечения $\{L_w\}$ и принципом оптимизации по направлению в n -мерном единичном кубе позволило разработать точные алгоритмы решения задач дискретного программирования, в общем случае имеющие экспоненциальную сложность, но работающие в среднем с полиномиальной сложностью при решении задач, содержащих до нескольких сотен переменных.

Показана возможность применения метода решения оптимизационных задач с БП на основе рангового подхода к решению ЗНП. Поскольку ЗНП полиномиально сводится к задачам: «наименьшее доминирующее множество вершин в графе»; «независимое наибольшее множество вершин в графе»; «наибольшая клика в графе»; «наибольшее паросочетание в графе», то рассмотренные приближенные и точные алгоритмы решения ЗНП могут быть использованы и для решения перечисленных задач, имеющих широкое прикладное значение в теории сложных систем.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Авдеева Е.С. Модели аналитической подготовки проекта внедрения корпоративной информационной системы на предприятии: дис. ... канд. техн. наук. Владимир, 2012. 153
2. Аверченков, В.И. Анализ и некоторая классификация методов доступа к данным / Аверченков В.И. и др. // Вестник компьютерных и информационных технологий. 2014. № 12. С. 48 - 56.
3. Божко В.П., Власов Д.В., Гаспариан М.С. Информационные технологии в экономике и управлении: Учебно-методический комплекс. – М.: Изд. центр ЕАОИ. 2010. – 120 с.
4. Бабиков А.Ю., Кардаш Д.И. Особенности использования мобильных устройств в работе защищенных корпоративных информационных систем // Вестник Уфимского государственного авиационного технического университета. 2013. Т. 17. №2(55). С. 157-162.
5. Березин Д.А. Оптимизация процесса внедрения корпоративной информационной системы на предприятии: дис. ... канд. экон. наук. Екатеринбург, 2003. 177 с.
6. Бова В.В., Лещанов Д.В., Кравченко Д.Ю. ИТ-архитектура корпоративных систем управления знаниями: интеграция программных решений, сервисов и унаследованных приложений // Информатика, вычислительная техника и инженерное образование. 2016. № 1 (25). С. 27-39.
7. Бураков П.В., Корпоративные информационные системы. Учебное пособие. - СПб НИУ ИТМО, 2014. – 96с.
8. Бушуева Л.И. Проблемы внедрения корпоративных информационных систем // Корпоративное управление и инновационное развитие экономики Севера: Вестник Научно-исследовательского центра корпоративного права, управления и венчурного инвестирования Сыктывкарского государственного университета. 2005. №3. С. 62-73.

9. Буй, Нгок Зыонг Архитектура системы управления мобильностью в корпоративной сети // Журнал «Современная наука: актуальные проблемы теории и практики». Сер. Естественные и технические науки - 2016. - № 09-10. - С. 30-35.

10.Буй, Нгок Зыонг Информационная безопасность процесса регистрации Android-устройства в системе управления корпоративной мобильностью / Нгок Зыонг Буй, А.Г. Кравец, Ле Тхань Тунг Нгуен // Вестник компьютерных и информационных технологий. - 2016. - № 8. - С. 44-51.

11.Веловатый Е.А. Оптимизация корпоративных информационных систем с использованием методов тензорного анализа: дис. ... канд. техн. наук. Екатеринбург, 2015. 135 с.

12.Вишняков В.А. Использование интеллектуальных и блокчейн технологий в информационном управлении // Системный анализ и прикладная информатика. 2018. № 1. С. 45-50.

13.Дли М.И. Создание информационных систем управления интегрированными предприятиями / М.И. Дли, В.В. Голубовская, В.Ю. Кинелев, В.А. Шумаев: Монография. – М.: Издательский дом «Экономическая газета», 2013. – 168с.

14.Древинг С., Такатлы Д. Методики расчетов экономической эффективности корпоративных информационных систем // РИСК: Ресурсы, информация, снабжение, конкуренция. 2011. №2. С. 581-585.

15.Жевнерчук Д.В. Анализ, принципы технической самоорганизации и структурно-параметрический синтез открытых информационных систем: дис. ... доктора техн. наук. Нижний Новгород, 2018. 317 с.

16.Зараменских Е.П. Управление жизненным циклом информационных систем: монография / Е.П. Зараменских. – Новосибирск: Издательство ЦРНС, 2014. – 270 с.

17.Зеленков Ю.А. Количественный анализ эффективности и риска внедрения информационных систем // Управление большими системами: сборник трудов. 2016. №60. С. 63-81

18. Зеленков Ю.А. Методология стратегического управления развитием корпоративной информационной системы крупного промышленного предприятия в современных условиях: дис. ... доктора техн. наук. Челябинск, 2013. 275 с.

19. Зотова А.С. Совершенствование процессов управления формированием и развитием корпоративных информационных систем промышленных предприятий: дис. ... канд. экон. наук. Самара, 2012. 162 с.

20. Зыков С.В. Технология интеграции данных в гетерогенных корпоративных программных комплексах: дис. ... доктора техн. наук. Москва, 2017. 128 с.

21. Вишняков В.А., Бородаенко Ю.В. Интегрированные кис как основа современного управления предприятием // Инновационные образовательные технологии. 2013. №1(33). С. 62-68.

22. Кокунов В.А., Минков В.И., Соколов Н.Е Особенности выбора корпоративной информационной системы (КИС) для предприятий горного дела // Вестник Брянского государственного университета. 2014. №3. С. 56-63.

23. Копытов В.В., Шульгин А.О., Федоров С.А. Разработка архитектуры интеграционной среды кроссплатформенных мобильных приложений с корпоративной информационной системой // Современные информационные технологии и ИТ-образование. 2015. Т. 11. №2. С. 71-77.

24. Корпоративные информационные системы управления: учебник / ред.: Н. М. Абдиев, О. В. Китова. - М.: ИНФРА-М, 2015. - 464 с.

25. Корпоративные информационные системы: учеб. пособие / В.А. Погонин, А.Г. Схиртладзе, С.И. Татаренко, С.Б. Путин. – Тамбов: Изд-во ФГБОУ ВПО «ТГТУ», 2012. – 144 с.

26. Козунова С.С., Бабенко А.А. Модель построения защищенной информационной системы корпоративного типа // Информационные системы и технологии. 2016. №3(95). С. 112-120

27. Куприянов Ю.В. Разработка формализованного подхода к управлению организационными изменениями при внедрении информационных систем на промышленных предприятиях: дис. ... канд. техн. наук. Краснодар, 2014. 186 с.

28.Куприянов Ю.В. Формализованный подход управления организационной трансформацией при внедрении информационной системы на промышленном предприятии / Ю.В. Куприянов, В.В. Таратухин // Научнотехнические ведомости СПбГПУ. - 2013. - Вып. 1. - С. 93-98

29.Куприянов, Ю.В. Принципы управления выгодами на проектах внедрения корпоративных информационных систем / Ю.В. Куприянов, В.В. Таратухин// Информационные технологии. - 2011. - Вып. 6. - С.66-71.

30.Курейчик В.В., Бова В.В., Лещанов Д.В. Интеграция проектных решений в корпоративных прикладных средах // Информационные технологии в науке, образовании и управлении Материалы XLIV международной конференции и XIV международной конференции молодых учёных ИТ + S&E`16. Под редакцией Е.Л. Глориозова. 2016. С. 221-228.

31.Ланчаков А.Б. Управление промышленным предприятием в условиях неопределенности на основе формирования адаптивных структур: дис. канд. экон. наук. Тула, 2018. 155 с.

32.Масленникова О.Е., Назарова О.Б. Типовой проект внедрения корпоративной информационной системы для строительных организаций // Математическое и программное обеспечение систем в промышленной и социальной сферах. 2015. №1(6). С. 54-60

33.Масленникова О.Е., Назарова О.Б. Типовой проект внедрения корпоративной информационной системы для строительных организаций // Электротехнические системы и комплексы. 2015. №2(27). С. 47-52.

34.Нартикоев А.А., Гугкаева К.В., Волик М.В. Особенности управления внедрением информационных систем на предприятии // Экономика и социум. 2017. №4(35). С. 1890-1895.

35.Нурдинов Р.А., Каторин Ю.Ф., Зайцева Н.М. Модель количественной оценки рисков безопасности информационной системы // Новый университет. Серия: Технические науки. 2016. № 3 (49). С. 42-47

36.Нурдинов Р.А. Модель количественной оценки рисков безопасности корпоративной информационной системы на основе метрик: дис. ... канд. техн. наук. Санкт-Петербург, 2016. 186 с.

37.Одинаец М.А. Перспективы развития ERP-систем // Экономика и управление №114. - 2014 – С. 73-77

38.Олейник П.П. Корпоративные информационные системы Учебник для вузов. Стандарт третьего поколения. - СПб.: Питер, 2012. - 176 с.

39.Мезенцев К.Н. Автоматизированные информационные системы. – М.: Академия, 2012. – 174 с.

40.Орлов, С. Корпоративная мобильность: что и как защищаем [Электронный ресурс] / С. Орлов // Журнал сетевых решений LAN. - 2014, №04. - Режим доступа: <http://www.osp.ru/lan/2014/04/13040708/> (дата доступа: 05.11.2018).

41.Павлов Н. Применение модульных решений для автоматизированных информационных систем / Павлов Н. // Современные технологии автоматизации. 2013. Т. 1. С. 62-68.

42.Пирогов В.Ю. Информационные системы и базы данных. Организация и проектирование. – СПб.: БХВ-Петербург, 2011. – 528 с.

43.Плетнев А.А. Динамическая база данных, допускающая параллельную обработку произвольных потоков запросов// Интеллектуальные системы. Теория и приложения. - 2015. Т. 19, Вып. 1. - С. 117–142.

44.Плетнев А.А. Информационно-графовая модель динамических баз данных и ее применение// Интеллектуальные системы. Теория и приложения. - 2014. Т. 18,Вып. 1. - С. 111-140.

45.Разумников С.В. Анализ существующих методов оценки эффективности информационных технологий для облачных ИТ-сервисов [Электронный ресурс] // Современные проблемы науки и образования. - 2013 - №. 3. - С. 1. - Режим доступа: www.science-education.ru/109-9548.

46.Разумников С.В. Модели поддержки принятия решений при выборе облачных ИТ-сервисов для внедрения на предприятии: дис. ... канд. техн. наук. Томск, 2016. 158 с.

47.Рудакова Е.В. Практика управления информационными ресурсами в корпоративных информационных системах // Актуальные проблемы социально-гуманитарного и научно-технического знания. 2016. №3(8). С. 1-2.

48.Руденко М.В. Выбор оптимальной стратегии синтеза модульных систем обработки данных для биллинговых мобильных приложений [Электронный ресурс] / Руденко М.В. // «Научные труды КубГТУ». - Краснодар: КубГТУ. - 2014. - №6. - Режим доступа: <http://ntk.kubstu.ru/file/289>.

49.Свид. о гос. регистрации программы для ЭВМ № 2016615613 от 26 мая 2016 г. Российская Федерация, МПК (нет). Модуль управления устройствами в корпоративной мобильной сети / А.Г. Кравец, Нгок Зыонг Буй, Ле Тхань Тунг Нгуен; ВолгГТУ. - 2016.

50.Третьяк В.Ф., Пашнева А. А. Оптимизация структуры хранилища данных в узлах сети инфокоммуникационной сети облачного хранилища // Системы навигации и связи Полтавского национального университета имени Юрия Кондратюка. №4 (44). - 2017 - С. 122-128

51.Campagna, R. Mobile Device Security for Dummies / Rich Campagna, Iyer Subbu, Krishnan Ashwin. - Wiley Publishing, Inc., 2011. - 294 с.

52.Listrovoy, S.V., Tretjak, V.F. and Listrovaya, A.S. (1999), “Parallel algorithms of calculation process optimization for the Boolean programming problems”, Ibid., Vol. 16, pp. 569-579

53.Enterprise Mobility Management - Market Quadrant 2016 // The Radicati Group, Inc. - 2016. – P. 42.

54.Straus, S. G. Mobile Technology and Action Teams: Assessing BlackBerry Use in Law Enforcement Units / S. G. Straus, T. K. Bikson, E. Balkovich, J. F. Pane // Computer Supported Cooperative Work (CSCW). – 2010. – С. 45-71.

55.Fowler, M. Patterns of Enterprise Application Architecture / M. Fowler. - Wiley Publishing, Inc., 2012. – 560 с.

ПРИЛОЖЕНИЕ А

Сопровождение и развитие систем 1С



ПРИЛОЖЕНИЕ Б

Решение намеченных задач

Растущие требования к автоматизации и контролю



1С: Охрана окружающей среды



1С: Управление холдингом



Внедрение на pilotных предприятиях



Тиражирование на предприятия



ПРИЛОЖЕНИЕ В

Листинг программы

```
//-----  
  
#include <vcl.h>  
#pragma hdrstop  
  
#include "Unit_head.h"  
#include "Unit_graph.h"  
#include "Unit_test.h"  
//-----  
#pragma package(smart_init)  
#pragma link "RzPanel"  
#pragma link "RzButton"  
#pragma link "RzCmboBx"  
#pragma link "RzLabel"  
#pragma link "RzShellDialogs"  
#pragma link "RzEdit"  
#pragma resource "* .dfm"  
TForm_Head *Form_Head;  
//-----  
__fastcall TForm_Head::TForm_Head(TComponent* Owner)  
    : TForm(Owner)  
{  
}  
}  
//-----  
//----- Generate Multigraph ---  
void TForm_Head::GenerateMultigraph(int m,int n,double& a,vv& _B)  
{  
    /* инициализация матриц инцидентности */  
    vector<int> _vector(n+1,0);  
    _B.clear();  
    for(int i=0;i<=m;i++) _B.push_back(_vector);  
    int edge=0;//количество сгенерированных ребер  
    /* генерация минимально возможного числа ребер */  
    int col=0,next_col=0,row=0;  
    set<int,less<int>> setm,setn;  
    Randomize;  
    col=Rand(n,setn);  
    while(setm.size()!=m)  
    {  
        row=Rand(m,setm);//случайная генерация новой строки  
        if(setn.size()!=n)  
        {  
            next_col=Rand(n,setn);//случайная генерация нового столбца  
            _B[row][col]=1;  
            _B[row][next_col]=1;  
            _B[0][col]++;  
            _B[0][next_col]++;  
            col=next_col;  
            edge+=2;//подсчет сгенерированных ребер  
        }  
    }  
}
```

```

    }
else
{
do
{
    col=Ceil(random(n)+1);
    next_col=Ceil(random(n)+1);
}
while(col==next_col);

    _B[row][col]=1;
    _B[row][next_col]=1;
    _B[0][col]++;
    _B[0][next_col]++;
    edge+=2;//подсчет сгенерированных ребер
}
}//while(setm.size()!=m)

while(setn.size()!=n)
{
    row=Ceil(random(m)+1);
    col=Rand(n,setn);
    _B[row][col]=1;
    _B[0][col]++;
    edge++;
}
/* дальнейшая генерация ребер до заданной плотности */
while(a>(edge,double(m*n)))
{
do
{
    row=Ceil(random(m)+1);
    col=Ceil(random(n)+1);
}
while(_B[row][col]==1);

    _B[row][col]=1;
    _B[0][col]++;
    edge++;//подсчет сгенерированных ребер
}
a=edge,double(m*n);
}
//----- end Generate Multigraph ---
//----- Generate Graph ---
void TForm_Head::GenerateGraph(int n,double& _A,vv& _W)
{
    /* инициализация матрицы смежности */
    vector<int> _vector(n+1,0);
    _W.clear();
    for(int i=0;i<=n;i++)_W.push_back(_vector);

    /* задание промежуточных переменных */

```

```

int vert=0,next_vert=0,edge=0;
set<int,less<int> > setn;

vert=Rand(n,setn);
for(int i=1;i<=n-1;i++)
{
    next_vert=Rand(n,setn);
    _W[vert][next_vert]=1;
    _W[next_vert][vert]=1;
    vert=next_vert;
    edge++; //подсчет сгенерированных ребер
}
while(_A>(edge,double(n*(n-1)/2)))
{
    do
    {
        vert=Ceil(random(n)+1);
        next_vert=Ceil(random(n)+1);
    }
    while((_W[vert][next_vert]==1)|| (vert==next_vert));

    _W[vert][next_vert]=1;
    _W[next_vert][vert]=1;
    edge++; //подсчет сгенерированных ребер
}

_A=edge,double(n*(n-1)/2);
}
//----- end Generate Graph ---
//----- Generate Tree ---
void TForm_Head::GenerateTree(int m,int n,vv<> _W)
{
    /* инициализация матрицы смежности */
    vector<int> _vector(n+1,0);
    _W.clear();
    for(int i=0;i<=n;i++) _W.push_back(_vector);
    set<int,less<int> > setn,tier,tier_next;
    int vertex=0,vertex_next=0,vert_on_tier=0;
    Randomize;
    vertex=Rand(n,setn);
    tier.insert(vertex);

    vert_on_tier=1;

    for(int i=1;i<m;i++)
    {
        vert_on_tier=Ceil(random((n-setn.size())-(m-i))+1);
        tier_next.clear();
        for(int j=1;j<=vert_on_tier;j++)
        {
            vertex=Rand(n,setn);
            tier_next.insert(vertex);
        }
    }
}

```

```

        }
        set<int,less<int> > copytier_next(tier_next);
        do
        {
            vertex=Rand(tier);
            vertex_next=Rand(copytier_next);
            copytier_next.erase(vertex_next);
            _W[vertex][vertex_next]=1;
            _W[vertex_next][vertex]=1;
        }
        while(copytier_next.size()!=0);

        tier.clear();
        tier=tier_next;
    }
    tier_next.clear();
    int lenght=n-setn.size();
    for(int i=1;i<=lenght;i++)
    {
        vertex=Rand(n,setn);
        tier_next.insert(vertex);
    }
    do
    {
        vertex=Rand(tier);
        vertex_next=Rand(tier_next);
        tier_next.erase(vertex_next);
        _W[vertex][vertex_next]=1;
        _W[vertex_next][vertex]=1;
    }
    while(tier_next.size()!=0);
}
//----- Generate Tree ---
//----- rand ---
int TForm_Head::Rand(int range,s& set_)
{
    int lenght=set_.size(),number=0;

    do
    {
        number=Ceil(random(range)+1);
        set_.insert(number);
    }
    while(lenght==set_.size());

    return number;
}
//----- end Rand ---
//----- rand ---
int TForm_Head::Rand(s set_)
{
    int lenght=set_.size(),number=0;

```

```

set<int,less<int>>::iterator it;

number=Ceil(random(length));
it=set_.begin();
for(int i=1;i<=number;i++)it++;

return *it;
}
//----- end Rand ---
//----- TransformWToB ---
void TForm_Head::TransformWToB(vv w,vv& _B)
{
    /* преобразование смежной матрицы в прямоугольную */
vector<int> _vector(w.size(),0);
_B.clear();
_B.push_back(_vector);

for(int i=1;i<w.size()-1;i++)
{
for(int j=i+1;j<w.size();j++)
if(w[i][j]==1)
{
    _B.push_back(_vector);
    _B[_B.size()-1][i]=1;
    _B[_B.size()-1][j]=1;
    _B[0][i]++;
    _B[0][j]++;
}
}
}
//----- TransformWToB ---
//----- TransformToE ---
void TForm_Head::TransformToE(vv b,vs& _E)
{
    /* преобразование матрицы инцидентности в таблицу строки в вершинах */
set<int,less<int>> _set;
_E.clear();
for(int i=0;i<b[0].size();i++)_E.push_back(_set);

for(int i=1;i<b.size();i++)
{
for(int j=1;j<b[0].size();j++)
if(b[i][j]==1)_E[j].insert(i);
}
Caption="GraphLAB";
}
//----- TransformToE ---
//----- TransformToD ---
void TForm_Head::TransformToD(vv w,vs& _D)
{
    /* преобразование прямоугольной матрицы в обратную и десятичную */
set<int,less<int>> _set;

```

```

    _D.clear();
    for(int i=0;i<w.size();i++)_D.push_back(_set);

    for(int i=1;i<w.size();i++)
    {
        for(int j=1;j<w[0].size();j++)
        if(w[i][j]==1)_D[i].insert(j);
    }
}

//----- TransformToD ---
//----- TransformEToW ---
void TForm_Head::TransformEToW(vs e,vv& _W)
{
    /* преобразование обратной матрицы в смежную */
    vector<int> _vector(e.size(),0);
    set<int,less<int> > _set;
    _W.clear();
    for(int i=0;i<e.size();i++)_W.push_back(_vector);

    int length=0;
    for(int i=1;i<e.size()-1;i++)
    {
        for(int j=1+1;j<e.size();j++)
        {
            _set.clear();
            set_union(e[i].begin(),e[i].end(),e[j].begin(),e[j].end(),
                      inserter(_set,_set.begin()));
            length=e[i].size()+e[j].size();
            if(length!=_set.size())
            {
                _W[i][j]=1;
                _W[j][i]=1;
            }
        }
    }
}

//----- TransformEToW ---
//----- SearchCoverRM ---
void TForm_Head::SearchCoverRM(vv b,vs& _COVER)
{
    _COVER.clear();

    /* инициализация упорядоченного вектора столбцов B(p)<=B(p+1) */
    pair<int,int> _pair(0,0);
    vector<pair<int,int> > I(b[1].size(),_pair);
    for(int i=1;i<b[1].size();i++)
    {
        I[i].first=b[0][i];
        I[i].second=i;
    }
    sort(I.begin(),I.end());

    /* формирование калибровочного вектора H(i) */
}

```

```

vector<int> vector1(b.size(),0);
vector<vector<int>> h(b[1].size(),vector1);// инициализация калибровочного вектора n*m

vector<vector<int>> vector2(2,vector1);// инициализация множества объединений
калибровочных векторов
vector<vector<v>> way(b[1].size(),vector2);// H(p)UH(s) дерева всех путей

set<int,less<int>> _set;
vector<set<int,less<int>>> vector_set(2,_set);
vector<vector<s>> cover(b[1].size(),vector_set);// инициализация множества всех путей
int value=0;
int number_uncover=0;//количество непокрытых строк

for(int i=1;i<b.size();i++)
{
    if(!b[i][I[b[1].size()-1].second])
    {
        h[b[1].size()-1][i]=9999;
        way[b[1].size()-1][1][i]=9999;
        number_uncover++;
    }
}

if(number_uncover==0)
{
    set<int,less<int>> _set;
    _set.insert(I[b[1].size()-1].second);
    _COVER.push_back(_set);
    return;
}                                // SearchCover RM

number_uncover=0;
for(int i=b[1].size()-2;i>=1;i--)
{
    for(int j=1;j<b.size();j++)
    {
        if(b[j][I[i].second])value=0;
        else if(h[i+1][j]==9999)value=9999;
        else value=h[i+1][j]+1;
        h[i][j]=value;
        way[i][1][j]=value;
        if(value!=0)number_uncover++;
    }
}

if(number_uncover==0)
{
    set<int,less<int>> _set;      // SearchCover RM
    _set.insert(i);
    _COVER.push_back(_set);
    return;
}

for(int i=1;i<cover.size();i++)cover[i][1].insert(I[i].second);

/* поиск пути с d=0 */

```

```

int dmin=9999;
for(int r=1;r<b[1].size();r++)
{
    SearchCoverRM(r,dmin,b,_COVER,way,way,cover,cover,h,I);
    if(_COVER.size()>1) return;
}
}

//----- end SearchCoverRM ---
//----- Rangs method ---
void TForm_Head::SearchCoverRM(int r,int& _dmin,vv b,vs& _COVER,vvv way,vvv& _way,vvs
cover,vvs& _cover,vv h,vp I)
{
    vector<int> _vector(b.size(),0);
    set<int,less<int> > cov;
    set<int,less<int> > d;
    int value=0;
    int number_uncover=0;//количество непокрытых строк
    int dmin=9999;

    for(int i=1;i<=way.size()-1;i++) _way[i].clear();
    for(int i=1;i<=cover.size()-1;i++) _cover[i].clear();

    for(int i=r+1;i<b[1].size();i++)
    {
        int _k=0;
        _way[i].push_back(_vector);
        _cover[i].push_back(cov);           // SearchCover RM
        for(int j=r;j<i;j++)
        {
            for(int k=1;k<way[j].size();k++)
            {
                if(way[j][k][0]<=_dmin)
                {
                    _k++;
                    _way[i].push_back(_vector);
                    _cover[i].push_back(cover[j][k]);
                    _cover[i][_k].insert(I[i].second);
                    number_uncover=0;
                    for(int l=1;l<way[j][k].size();l++)
                    {
                        if(way[j][k][l]!=0)
                        {
                            value=h[i][l];
                            if(h[i][l]!=0)number_uncover++;
                        }
                        else value=0;
                        if(value==9999)      // удаление пути не удовлетворяющих условию d!=9999
                        {
                            _way[i].pop_back();
                            _cover[i].pop_back();
                            number_uncover=1;
                            _k--;
                        }
                    }
                }
            }
        }
    }
}

```

```

        break;
    }
    _way[i][_k][l]=value; // SearchCover RM
    d.insert(value);
}

if(number_uncover==0)
{
    set<int> _set;
    set<int,less<int> >::iterator it;
    for(it=_cover[i][_k].begin();it!=_cover[i][_k].end();it++)
    {
        _set.clear();
        _set.insert(*it);
        _COVER.push_back(_set);
    }
    return;
}

if(value!=9999)
{
    int posibiliti_cover=0;
    int d0=0;
    int k=b[0].size()-1;
    do
    {
        d0++;
        posibiliti_cover=posibiliti_cover+b[0][I[k].second];
        k--;
    }
    while(posibiliti_cover<number_uncover);
    _way[i][_k][0]=d0;

    if(dmin>d.size()-1)dmin=d.size()-1;
    d.clear();
}
}//----- if way[j][k][0]<=d_min
}//----- for k
}//----- for j
}
_dmin=dmin;
}
//----- end SearchCoverRM ---
//----- SearchCoverRM X ---
void TForm_Head::SearchCoverRM(double& _X,vv b,vs& _COVER)
{
    _COVER.clear();
    /* инициализация упорядоченного вектора столбцов B(p)<=B(p+1) */
    pair<int,int> _pair(0,0);
    vector<pair<int,int> > I(b[1].size(),_pair);
    for(int i=1;i<b[1].size();i++)
    {

```

```

I[i].first=b[0][i];
I[i].second=i;
}
sort(I.begin(),I.end());

/* формирование калибровочного вектора H(i) */
vector<int> vector1(b.size(),0);
vector<vector<int> > h(b[1].size(),vector1);// инициализация калибровочного вектора n*m

vector<vector<int> > vector2(2,vector1);// инициализация множества объединений
калибровочных векторов
vector<vector<v> > way(b[1].size(),vector2);// H(p)UH(s) дерева всех путей

set<int,less<int> > _set;
vector<set<int,less<int> > > vector_set(2,_set);
vector<vector<s> > cover(b[1].size(),vector_set);// инициализация множества всех путей

int value=0;
int number_uncover=0;//количество непокрытых строк

for(int i=1;i<b.size();i++)
    if(!b[i][I[b[1].size()-1].second])           // SearchCover RM X
    {
        h[b[1].size()-1][i]=9999;
        way[b[1].size()-1][1][i]=9999;
        number_uncover++;
    }

if(number_uncover==0)
{
    set<int,less<int> > _set;
    _set.insert(I[b[1].size()-1].second);
    _COVER.push_back(_set);
    return;
}

number_uncover=0;
for(int i=b[1].size()-2;i>=1;i--)
{
    for(int j=1;j<b.size();j++)
    {
        _X++;                                // SearchCover RM X

        if(b[j][I[i].second])value=0;
        else if(h[i+1][j]==9999)value=9999;
            else value=h[i+1][j]+1;
        h[i][j]=value;
        way[i][1][j]=value;
        _X++;
        if(value!=0)number_uncover++;
    }
}

if(number_uncover==0)
{
    set<int,less<int> > _set;
}

```

```

        _set.insert(i);
        _COVER.push_back(_set);
        return;
    } }
for(int i=1;i<cover.size();i++)cover[i][1].insert(I[i].second);
/* поиск пути с d=0 */
int dmin=99999;
for(int r=1;r<b[1].size();r++)
{
    SearchCoverRM(_X,r,dmin,b,_COVER,way,way,cover,cover,h,I);
    if(_COVER.size()>1)return;
}
}
//----- end SearchCoverRM X ---
//----- SearchCoverRM X ---
void TForm_Head::SearchCoverRM(double& _X,int r,int& _dmin,vv b,vs& _COVER,vvv way,vvv&
_way,vvs cover,vvs& _cover,vv h,vp I)
{
    vector<int> _vector(b.size(),0);
    set<int,less<int> > cov;
    set<int,less<int> > d;
    int value=0;
    int number_uncover=0;//количество непокрытых строк
    int dmin=9999;
    for(int i=1;i<=way.size()-1;i++)_way[i].clear();
    for(int i=1;i<=cover.size()-1;i++)_cover[i].clear();
    for(int i=r+1;i<b[1].size();i++)
    {
        int _k=0;
        _way[i].push_back(_vector);
        _cover[i].push_back(cov);
        for(int j=r;j<i;j++)
        {
            for(int k=1;k<way[j].size();k++)
            {
                _X++; // SearchCover RM X
                if(way[j][k][0]<=_dmin)
                {
                    _k++;
                    _way[i].push_back(_vector);
                    _cover[i].push_back(cover[j][k]);
                    _cover[i][_k].insert(I[i].second);
                    number_uncover=0;
                    for(int l=1;l<way[j][k].size();l++)
                    {
                        _X++;
                        if(way[j][k][l]!=0)
                        {
                            value=h[i][l];
                            if(h[i][l]!=0)number_uncover++;
                        }
                        else value=0;
                    }
                }
            }
        }
    }
}

```

```

if(value==9999)      // удаление пути не удовлетворяющих условию d!=9999
{
    _way[i].pop_back();
    _cover[i].pop_back();
    number_uncover=1;
    _k--;
    break;           // SearchCover RM X
}
_way[i][_k][l]=value;
d.insert(value);
}
_X++;
if(number_uncover==0)
{
set<int> _set;
set<int,less<int>>::iterator it;
for(it=_cover[i][_k].begin();it!=_cover[i][_k].end();it++)
{
    _set.clear();
    _set.insert(*it);
    _COVER.push_back(_set);
}
return;           // SearchCover RM X
}
_X++;
if(value!=9999)
{
    int possiliti_cover=0;
    int d0=0;
    int k=b[0].size()-1;
    do
    {
        d0++;
        possiliti_cover=possiliti_cover+b[0][I[k].second];
        k--;
    }
    while(possiliti_cover<number_uncover);
    _way[i][_k][0]=d0;

    if(dmin>d.size()-1)dmin=d.size()-1;
    d.clear();
}
}//----- if way[j][k][0]<=d_min
}//----- for k
}//----- for j
} _dmin=dmin;
}
//----- end SearchCoverRM X ---
//----- SearchCoverOM ---
void TForm_Head::SearchCoverOM(vv b,vs e,vs& _COVEROM)
{
_COVEROM.clear();

```

```

set<int,less<int> > _set,vertex;
vector<set<int,less<int> >> ways(b[0].size(),_set);
vector<set<int,less<int> >> lengths(b[0].size(),_set);
for(int i=1;i<ways.size();i++)
{
    vertex.insert(i);
    ways[i].insert(i);
    set<int,less<int> ::iterator it;
    for(it=e[i].begin();it!=e[i].end();it++)
        lengths[i].insert(*it);
}
for(int r=1;r<b[0].size();r++)
{
    SearchCoverOM(b.size()-1,ways,lengths,ways,lengths,vertex,e,_COVEROM);
    if(_COVEROM.size()>0) return;
}
}
//----- end SearchCoverOM ---
//----- SearchCoverOM ---
void TForm_Head::SearchCoverOM(int m,vs ways,vs lengths,vs& _ways,vs& _lengths,s vertex,vs e,vs& _COVEROM)
{
    int number_max,length_max;
    set<int,less<int> > _set;
    set<int,less<int> ::iterator it;
    _ways.clear();
    _lengths.clear(); // OWN MAX
    _ways.push_back(_set);
    _lengths.push_back(_set);

    for(int i=1;i<ways.size();i++)
    {
        vertex.erase(i);
        number_max=0;
        length_max=0;
        for(it=vertex.begin();it!=vertex.end();it++)
        {
            _set.clear();
            set_union(lengths[i].begin(),lengths[i].end(),e[*it].begin(),e[*it].end(),
                      inserter(_set,_set.begin()));
            if(_set.size()>length_max)
            {
                length_max=_set.size(); // OWN MAX
                number_max=*it;
                if(length_max==m)
                {
                    set<int,less<int> ::iterator _it;
                    for(_it=ways[i].begin();_it!=ways[i].end();_it++)
                    {
                        set<int,less<int> __set;
                        __set.insert(*_it);
                        _COVEROM.push_back(__set);
                    }
                }
            }
        }
    }
}

```

```

        }
set<int,less<int>> __set;
__set.insert(number_max);
_COVEROM.push_back(__set);
return;
}
}// end if (_set.size()>length_max)
}// end for(it=vertex.begin();it!=vertex.end();it++)

/* формирование нового пути и его длины */
__set.clear();
_lengths.push_back(__set);
set_union(lengths[i].begin(),lengths[i].end(),e[number_max].begin(),e[number_max].end(),
    inserter(_lengths[i],_lengths[i].begin()));

_ways.push_back(ways[i]);
_ways[i].insert(number_max);
vertex.insert(i);
}// end for i
}
//----- end SearchCoverOM ---
//----- SearchCoverMEG ---
void TForm_Head::SearchCoverMEG(vs d,vs& _COVERMEG)
{
    _COVERMEG.clear();

    set<int,less<int>> vertex_set,cover_first;
    for(int i=1;i<d.size();i++)vertex_set.insert(i);
    set<int,less<int>> circl(vertex_set);
    int length=0;
    do
    {
        length=vertex_set.size();
        SearchExtrem(circl,vertex_set,cover_first,d);
    }
    while(length!=vertex_set.size());

    if(MaxDegree(vertex_set,d)<2)
    {
        _COVERMEG.push_back(cover_first);
        return;
    }
    set<int,less<int>>::iterator it;
    for(it=vertex_set.begin();it!=vertex_set.end();it++)
    {
        set<int,less<int>> cover_last(cover_first);
        SearchCoverMEG(*it,vertex_set,d,cover_last);
        _COVERMEG.push_back(cover_last);
    }
    SearchMinCover(_COVERMEG,_COVERMEG);}
//----- end SearchCoverMEG ---
//----- SearchCoverMEG ---
void TForm_Head::SearchCoverMEG(int vertex,s vertex_set,vs d,s& _cover)

```

```

{
    set<int,less<int> > circl;
    int length=0;
    ExcludeVertex(vertex,circl,vertex_set,d,_cover);

    do
    {
        do
        {
            length=vertex_set.size();
            SearchExtrem(circl,vertex_set,_cover,d);
        }
        while(length!=vertex_set.size());
        if(MaxDegree(vertex_set,d)<2) return;
        // поиск вершины добавление которой в МВП даст максимальное уменьшение
        // степеней всех вершин и добавление ее в МВП
        int best_vertex=VertexMaxSumDegree(vertex_set,d);

        ExcludeVertex(best_vertex,circl,vertex_set,d,_cover);
    }
    while(true);
}//----- end SearchCoverMEG ---
//----- SearchCoverMEG X ---
void TForm_Head::SearchCoverMEG(double& _X,vs d,s& _cover)
{
    _cover.clear();
    _X=0;
    set<int,less<int> > vertex_set,cover_first;
    for(int i=1;i<d.size();i++) vertex_set.insert(i);
    set<int,less<int> > cover_min(vertex_set);
    set<int,less<int> > circl(vertex_set);

    int length=0;
    do
    {
        length=vertex_set.size();
        SearchExtrem(_X,circl,vertex_set,cover_first,d);
    }
    while(length!=vertex_set.size());

    _X++;
    if(MaxDegree(_X,vertex_set,d)<2)
    {
        copy(cover_first.begin(),cover_first.end(),inserter(_cover,_cover.begin()));
        return;
    }
    set<int,less<int> ::iterator it;
    for(it=vertex_set.begin();it!=vertex_set.end();it++)
    {
        set<int,less<int> > cover_last(cover_first);
        SearchCoverMEG(_X,*it,vertex_set,d,cover_last);
        _X++;
        if(cover_min.size()>cover_last.size())

```

```

    {
        cover_min.clear();
        copy(cover_last.begin(),cover_last.end(),inserter(cover_min,cover_min.begin())));
    } }
copy(cover_min.begin(),cover_min.end(),inserter(_cover,_cover.begin())));
}
//----- end SearchCoverMEG X ---
//----- X SearchCoverMEG ---
void TForm_Head::SearchCoverMEG(double& _X,int vertex,s vertex_set,vs d,s& __cover)
{
    set<int,less<int> > circl;
    int length=0;
    ExcludeVertex(_X,vertex,circl,vertex_set,d,__cover);
    do
    {
        do
        {
            length=vertex_set.size();
            SearchExtrem(_X,circl,vertex_set,__cover,d);      }
            while(length!=vertex_set.size());
            _X++;
            if(MaxDegree(_X,vertex_set,d)<2)return;

            // поиск вершины добавление которой в МВП даст максимальное уменьшение
            // степеней всех вершин и добавление ее в МВП
            int best_vertex=VertexMaxSumDegree(_X,vertex_set,d);

            ExcludeVertex(_X,best_vertex,circl,vertex_set,d,__cover);
        }
        while(true);}
}
//----- end X SearchCoverMEG ---
//----- SearchMinCover ---
void TForm_Head::SearchMinCover(vs coverme,vs& _COVERME)
{
    _COVERME.clear();
    set<int,less<int> > min_cover;
    set<int,less<int> >::iterator it;
    int min_length=999999;
    for(int i=0;i<coverme.size();i++)
        if(min_length>=coverme[i].size())
    {   if(min_length>coverme[i].size())
        {
            min_cover.clear();
            min_length=coverme[i].size();
            min_cover.insert(i);
        }
        else min_cover.insert(i);
    }
    for(it=min_cover.begin();it!=min_cover.end();it++)
        _COVERME.push_back(coverme[*it]);
}
//----- end SearchMinCover ---
//----- SearchExtrem ---
void TForm_Head::SearchExtrem(s& _circl,s& _vertex_set,s& _cover,vs& _d)

```

```

{
    set<int,less<int> > buffer_circl;
    set<int,less<int> >::iterator it;

    for(it=_circl.begin();it!=_circl.end();it++)
    {
        if(_d[*it].size()==1)
        {
            //запоминаем вершину смежную с крайней
            int adjacent=*_d[*it].begin();

            //удаляем крайнюю вершину и ее связи
            _vertex_set.erase(*it);
            EraseVertex(*it,_d);

            //вносим смежную в ВП
            _cover.insert(adjacent);

            //создаем новое окружение в промежуточном контейнере удаленных вершин
            NewCircl(adjacent,buffer_circl,_d);

            //удаляем смежную вершину и ее связи
            _vertex_set.erase(adjacent);
            EraseVertex(adjacent,_d);
        }
    }

    //создаем новое окружение удаленных вершин
    _circl.clear();
    copy(buffer_circl.begin(),buffer_circl.end(),inserter(_circl,_circl.begin()));
}

//-----
//----- end SearchExtrem ---
//----- SearchExtrem X ---
void TForm_Head::SearchExtrem(double& _X,s& _circl,s& _vertex_set,s& _cover,vs& _d)
{
    set<int,less<int> > buffer_circl;
    set<int,less<int> >::iterator it;
    for(it=_circl.begin();it!=_circl.end();it++)
    {
        _X++;
        if(_d[*it].size()==1)
        {
            //запоминаем вершину смежную с крайней
            int adjacent=*_d[*it].begin();

            //удаляем крайнюю вершину и ее связи
            _vertex_set.erase(*it);
            EraseVertex(_X,*it,_d);

            //вносим смежную в ВП
            _cover.insert(adjacent);

            //создаем новое окружение в промежуточном контейнере удаленных вершин
            NewCircl(adjacent,buffer_circl,_d);
    }
}

```

```

//удаляем смежную вершину и ее связи
    _vertex_set.erase(adjacent);
    EraseVertex(_X,adjacent,_d);
}
}

//создаем новое окружение удаленных вершин
_circl.clear();
copy(buffer_circl.begin(),buffer_circl.end(),inserter(_circl,_circl.begin()));
}

//----- end SearchExtrem X ---
//----- ExcludeVertex ---
void TForm_Head::ExcludeVertex(int vertex,s& _circl,s& _vertex_set,vs& _d,s& _cover)
{
    set<int,less<int>>::iterator it;
    set<int,less<int>> adjacent;

    for(it=_d[vertex].begin();it!=_d[vertex].end();it++) adjacent.insert(*it);

    _circl.clear();
    for(it=adjacent.begin();it!=adjacent.end();it++)
    {
        _cover.insert(*it);
        _vertex_set.erase(*it);
        NewCircl(*it,_circl,_d);
        EraseVertex(*it,_d);
        _vertex_set.erase(vertex);
        EraseVertex(vertex,_d);
    }
}

//----- end ExcludeVertex ---
//----- X ExcludeVertex ---
void TForm_Head::ExcludeVertex(double& _X,int vertex,s& _circl,s& _vertex_set,vs& _d,s& _cover)
{
    set<int,less<int>>::iterator it;
    set<int,less<int>> adjacent;

    for(it=_d[vertex].begin();it!=_d[vertex].end();it++) adjacent.insert(*it);
    _circl.clear();
    for(it=adjacent.begin();it!=adjacent.end();it++)
    {
        _X++;
        _cover.insert(*it);
        _vertex_set.erase(*it);
        NewCircl(*it,_circl,_d);
        EraseVertex(_X,*it,_d);
    }
    _vertex_set.erase(vertex);
    EraseVertex(_X,vertex,_d);
}

//----- end X ExcludeVertex ---
//----- EraseVertex ---
void TForm_Head::EraseVertex(int vertex,vs& _d)
{
    set<int,less<int>>::iterator it;

    for(it=_d[vertex].begin();it!=_d[vertex].end();it++) _d[*it].erase(vertex);
    _d[vertex].clear();
}

```

```

}

//----- end EraseVertex ---
//----- X EraseVertex ---
void TForm_Head::EraseVertex(double& _X,int vertex,vs& _d)
{
    set<int,less<int>>::iterator it;

    for(it=_d[vertex].begin();it!=_d[vertex].end();it++)
    {
        _X++;
        _d[*it].erase(vertex);
    }
    _d[vertex].clear();
}

//----- end X EraseVertex ---
//----- NewCircl ---
void TForm_Head::NewCircl(int vertex,s& _circl,vs d)
{
    set<int,less<int>>::iterator it;
    for(it=d[vertex].begin();it!=d[vertex].end();it++) _circl.insert(*it);
}

//----- end NewCircl ---
//----- VertexMaxSumDegree ---
int TForm_Head::VertexMaxSumDegree(s vertex_set,vs d)
{
    //поиск вершины добавление которой в МВП даст максимальное уменьшение
    //степеней всех вершин
    set<int,less<int>>::iterator it;

    int max_degree=0;
    int vertex=*vertex_set.begin();

    for(it=vertex_set.begin();it!=vertex_set.end();it++)
        if(max_degree<SumDegree(*it,vertex_set,d))
        {
            vertex=*it;
            max_degree=SumDegree(*it,vertex_set,d);
        }
    return vertex;
}

//----- end VertexMaxSumDegree ---
//----- X VertexMaxSumDegree ---
int TForm_Head::VertexMaxSumDegree(double& _X,s vertex_set,vs d)
{
    //поиск вершины добавление которой в МВП даст максимальное уменьшение
    //степеней всех вершин
    set<int,less<int>>::iterator it;

    int max_degree=0;
    int vertex=*vertex_set.begin();

    for(it=vertex_set.begin();it!=vertex_set.end();it++)
    {
}

```

```

_X++;
if(max_degree<SumDegree(_X,*it,vertex_set,d))
{
    vertex=*it;
    max_degree=SumDegree(_X,*it,vertex_set,d);
}
return vertex;
}
//----- end X VertexMaxSumDegree ---


//----- SumDegree ---
int TForm_Head::SumDegree(int vertex,s vertex_set,vs d)
{
    int degree=0;
    set<int,less<int> ::iterator it;

    //умножение булевого выражения матрици смежности на *it_ вершину
    EraseVertex(vertex,d);
    vertex_set.erase(vertex);

    //вычисление суммарной степени всех вершин
    for(it=vertex_set.begin();it!=vertex_set.end();it++)
        degree+=d[*it].size();

    return degree;
}
//----- end SumDegree ---
//----- X SumDegree ---
int TForm_Head::SumDegree(double& _X,int vertex,s vertex_set,vs d)
{
    int degree=0;
    set<int,less<int> ::iterator it;

    //умножение булевого выражения матрици смежности на *it_ вершину
    EraseVertex(vertex,d);

    //вычисление суммарной степени всех вершин
    for(it=vertex_set.begin();it!=vertex_set.end();it++)
    {
        _X++;
        degree+=d[*it].size();
    }
    return degree;
}
//----- end X SumDegree ---
//----- MaxDegree ---
int TForm_Head::MaxDegree(s vertex_set,vs e)
{
    int max_degree=0;
    set<int,less<int> ::iterator it;

```

```

for(it=vertex_set.begin();it!=vertex_set.end();it++)
{
    if(max_degree<e[*it].size())max_degree=e[*it].size();
}
return max_degree;
}

//----- end MaxDegree ---
//----- MaxDegree X ---
int TForm_Head::MaxDegree(double& _X,s vertex_set,vs e)
{
    int max_degree=0;
    set<int,less<int> ::iterator it;

    for(it=vertex_set.begin();it!=vertex_set.end();it++)
    {
        _X++;
        if(max_degree<e[*it].size())max_degree=e[*it].size();
    }
    return max_degree;
}

//----- end MaxDegree X ---
//----- SearchCoverMEM ---
void TForm_Head::SearchCoverMEM(vs e,vs d,vs& _COVERMEM)
{
    _COVERMEM.clear();

    set<int,less<int> > col,row;
    //формирование множества всех столбцов
    for(int i=1;i<e.size();i++)col.insert(i);
    //формирование множества всех строк
    for(int i=1;i<d.size();i++)row.insert(i);
    set<int, less<int> ::iterator it;
    for(it=col.begin();it!=col.end();it++)
    {
        //поиск покрытия исключая i-ю вершину
        set<int, less<int> > cover;
        SearchCoverMEM(*it,col,row,d,e,cover);
        _COVERMEM.push_back(cover);
    }

    SearchMinCover(_COVERMEM,_COVERMEM);
}

//----- end SearchCoverMEM ---

//----- SearchCoverMEM ---
void TForm_Head::SearchCoverMEM(int number_col,s col,s row,vs d,vs e,s& _cover)
{
    _cover.insert(number_col);
    MultiplyOnCol(number_col,col,row,d,e);
    if(row.size()<1)return;
}

```

```

do
{
    // поиск столбца смежного с "крайним" столбцом
    bool seach;
    do
    {
        seach=false;
        set<int,less<int> > col_copy(col);
        set<int,less<int> ::iterator it_col;
        for(it_col=col_copy.begin();it_col!=col_copy.end();it_col++)
        {
            if(e[*it_col].size()==1)
                if(d[*e[*it_col].begin()].size()==2)
                {
                    d[*e[*it_col].begin()].erase(*it_col);
                    _cover.insert(*d[*e[*it_col].begin()].begin());
                    MultiplyOnCol(*d[*e[*it_col].begin()].begin(),col,row,d,e);
                    if(row.size()<1) return;
                    seach=true;
                }
            }
        }
        while(seach);

        // поиск лучшего столбца по заданному критерию
        int best_col=BestCol(col,row,d,e);

        _cover.insert(best_col);

        MultiplyOnCol(best_col,col,row,d,e);

        if(row.size()<1) return;
    }
    while(true);
}
//----- end SearchCoverMEM ---
//----- SearchCoverMEM X ---
void TForm_Head::SearchCoverMEM(double& _X,vs e,vs d,s& _cover)
{
    _cover.clear();
    _X=0;
    set<int,less<int> > col,row;
    //формирование множества всех столбцов
    for(int i=1;i<e.size();i++) col.insert(i);

    //формирование множества всех строк
    for(int i=1;i<d.size();i++) row.insert(i);

    //формирования начального минимального покрытия
    set<int,less<int> > cover_min(col);

    set<int, less<int> ::iterator it;

    for(it=col.begin();it!=col.end();it++)
    {

```

```

//поиск покрытия исключая i-ю вершину
set<int, less<int> > cover;

SearchCoverMEM(_X,*it,col,row,d,e,cover);
_X++;
if(cover_min.size()>cover.size())
{
    cover_min.clear();
    copy(cover.begin(),cover.end(),inserter(cover_min,cover_min.begin()));
}
copy(cover_min.begin(),cover_min.end(),inserter(_cover,_cover.begin()));
}

//----- end X SearchCoverMEM ---
//----- X SearchCoverMEM ---
/* поиск вершинного принимая что i-я вершина принадлежит минимальному ВП*/
void TForm_Head::SearchCoverMEM(double& _X,int number_col,s col,s row,vs d,vs e,s& __cover)
{
    _X++;
    __cover.insert(number_col);

    MultiplyOnCol(_X,number_col,col,row,d,e);

    if(row.size()<1)return;

    do
    { //поиск столбца смежного с "крайним" столбцом
        bool seach;
        do
        { seach=false;
            set<int,less<int> > col_copy(col);
            set<int,less<int> ::iterator it_col;
            for(it_col=col_copy.begin();it_col!=col_copy.end();it_col++)
            {
                _X++;
                if(e[*it_col].size()==1)
                {
                    _X++;
                    if(d[*e[*it_col].begin()].size()==2)
                    {
                        d[*e[*it_col].begin()].erase(*it_col);
                        __cover.insert(*d[*e[*it_col].begin()].begin());
                        MultiplyOnCol(_X,*d[*e[*it_col].begin()].begin(),col,row,d,e);
                        _X++;
                        if(row.size()<1)return;
                        seach=true;
                    }
                }
            }
        while(seach);
    }
}

```