

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Уфимский университет науки и технологий»

На правах рукописи



**БЕЖАЕВА Оксана Яковлевна**

**МЕТОДОЛОГИЧЕСКИЕ ОСНОВЫ ОБЕСПЕЧЕНИЯ  
ФУНКЦИОНАЛЬНОЙ НАДЕЖНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ  
НА ПРЕДПРОЕКТНОЙ СТАДИИ**

2.3.1. Системный анализ, управление и обработка информации, статистика  
(технические науки)

Диссертация на соискание ученой степени доктора технических наук

Научный консультант:  
доктор технических наук, профессор  
Гвоздев Владимир Ефимович

Уфа - 2025

**ОГЛАВЛЕНИЕ**

ВВЕДЕНИЕ .....	5
ГЛАВА 1. АНАЛИЗ ПРОБЛЕМНОЙ СИТУАЦИИ ОБЕСПЕЧЕНИЯ ФУНКЦИОНАЛЬНОЙ НАДЕЖНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ НА ПРЕДПРОЕКТНОЙ СТАДИИ.....	16
1.1    Актуальность проблемы обеспечения функциональной надежности информационных систем .....	17
1.2    Эволюция подходов к обеспечению функциональной надежности информационных систем .....	22
1.3    Анализ современного состояния исследований, связанных с обеспечением функциональной надежности информационных систем.....	27
1.4    Цель и задачи исследований .....	49
Выводы по главе 1 .....	52
ГЛАВА 2. КОНЦЕПТУАЛЬНЫЕ ОСНОВЫ ОБЕСПЕЧЕНИЯ ФУНКЦИОНАЛЬНОЙ НАДЕЖНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ .....	53
2.1    Принципы и подходы к обеспечению функциональной надежности информационных систем .....	54
2.2    Методология обеспечения функциональной надежности информационных систем на предпроектной стадии .....	62
2.3    Принципы исследования дефектов с позиции субъектоцентрической природы информационных систем .....	66
2.4    Системная модель формирования понятийного пространства, стратегии и внешнего облика информационной системы.....	71
2.5    Содержание системных архетипов в задачах обеспечения функциональной надежности информационных систем.....	78
Выводы по главе 2 .....	91
ГЛАВА 3. МОДЕЛИ ФАКТОРОВ, ПРИВОДЯЩИХ К ВОЗНИКНОВЕНИЮ ДЕФЕКТОВ И ОШИБОК НА ПРЕДПРОЕКТНОЙ СТАДИИ.....	93

3.1 Структурные модели процессов обеспечения функциональной надежности информационных систем .....	94
3.2 Формальная модель ошибок предпроектной стадии .....	113
3.3 Модель формирования интегрального показателя, характеризующего совокупное влияние факторов, как способствующих, так и препятствующих повышению функциональной надежности информационных систем.....	119
3.4 Динамические модели проблемных ситуаций на основе системных архетипов .....	127
Выводы по главе 3 .....	132
<b>ГЛАВА 4. МЕТОДИЧЕСКИЕ ОСНОВЫ ОБЕСПЕЧЕНИЯ ФУКЦИОНАЛЬНОЙ НАДЕЖНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ .....</b>	<b>134</b>
4.1 Метод построения многомерных зависимостей на основе совместного использования измерительных данных и экспертных оценок .....	135
4.2 Метод оценки функциональной надежности информационных сервисов на основе концепции профилей с учетом субъективных оценок пользователей .....	146
4.3 Метод структурного моделирования последствий отказов функциональных компонент информационных систем.....	153
4.4 Диверсионный анализ. Схема анализа событий в рамках AFD-1 .....	161
Выводы по главе 4 .....	169
<b>ГЛАВА 5. РАЗРАБОТКА ДЕСКРИПТИВНЫХ МОДЕЛЕЙ БАЗОВЫХ ХАРАКТЕРИСТИК ПРОЕКТА РАЗРАБОТКИ ИНФОРМАЦИОННЫХ СИСТЕМ .....</b>	<b>171</b>
5.1 Модель проекта разработки информационных систем .....	172
5.2 Формирование параметров модели управления проектом на основе линеаризации функциональных зависимостей.....	177
5.3 Оценка бюджета проекта по критериям удовлетворённости акторов .....	181
5.4 Построение моделей ситуаций, возникающих при реализации проектов.....	186
5.5 Оценка согласованности мнений неоднородных акторов на основе системных архетипов .....	202
Выводы по главе 5.....	209

ГЛАВА 6. ПРАКТИЧЕСКОЕ ИСПОЛЬЗОВАНИЕ НАУЧНЫХ РЕЗУЛЬТАТОВ В ЗАДАЧАХ ОБЕСПЕЧЕНИЯ ФУНКЦИОНАЛЬНОЙ НАДЕЖНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ.....	211
6.1 Выявление факторов функциональной надежности информационных систем при формировании единого информационного пространства компании.....	212
6.2 Методика формальной оценки возможности реализации проекта .....	227
6.3 Методика и программное обеспечение для анализа моделей проектируемых компонентов информационных систем по критериям функциональной надежности .....	243
Выводы по главе 6.....	265
ОСНОВНЫЕ РЕЗУЛЬТАТЫ И ВЫВОДЫ .....	266
ГЛОССАРИЙ.....	270
СПИСОК ЛИТЕРАТУРЫ.....	272
Приложение А .....	293
Приложение Б .....	298
Приложение В.....	311

## **ВВЕДЕНИЕ**

### **Актуальность темы исследования**

Базовым требованием к информационной инфраструктуре в условиях цифровой экономики является предоставление своевременной, полной и достоверной информации, необходимой для урегулирования ситуаций, возникающих в различных частях распределённой информационной системы. Согласно ГОСТ 33707-2016 [53], информационная система (ИС) - система, организующая обработку информации о предметной области и ее хранение.

Под функциональной надежностью ИС понимается свойство сохранять работоспособность в соответствии со своим целевым назначением при случайных дестабилизирующих воздействиях и отсутствии злоумышленного влияния на программную и аппаратную составляющую.

Функциональная надежность является латентной характеристикой ИС и объективно характеризуется удовлетворенностью потребителей способностью системы в нужное время предоставлять сведения, соответствующие ситуации, которую нужно урегулировать. Дефекты выступают в качестве критического фактора, определяющего функциональную надёжность. Источниками таких дефектов становятся ошибки, допускаемые субъектами на различных стадиях жизненного цикла ИС, причем дефекты предпроектной стадии в определении требований к свойствам ИС, являются причиной наиболее тяжелых, с точки зрения сетецентрического управления, последствий.

Современная практика показывает, что в области аппаратных компонентов накоплен значительный опыт обеспечения высоких характеристик надежности, при этом уровень надежности программных компонентов недостаточен.

Субъектоцентрическая природа проекта разработки ИС предопределяет то, что организационные, методические и технологические ошибки, допускаемые субъектами проектирования, являются неизменной составляющей любого проекта.

На это неоднократно указывалось в литературе и аналитических отчётах, в частности в исследованиях *Standish Group* [143-145]. Недостаточная надежность программных компонентов, в том числе, обусловлена недостаточным уровнем развития теоретической и методической базы управления состоянием ИС, в особенности на предпроектной стадии.

Рост масштабов и сложности ИС существенно ограничивает применимость традиционных подходов к обеспечению функциональной надёжности. В целом, к настоящему времени не сформированы теоретические основы построения систем обеспечения надёжности на предпроектной стадии.

**Степень разработанности темы.** Понятие «функциональная надежность» и «функциональная безопасность» до настоящего времени трактуется различными исследователями неоднозначно. Согласно положениям стандарта МЭК 61508-2-2012 [47], под функциональной безопасностью (*functional safety*) подразумевается корректное функционирование, как системы управления, так и управляемого оборудования. Липаев В.В. [73, 74, 76] в своих работах считает, что понятия и характеристики функциональной безопасности систем близки к понятиям надежности, определяет в соответствии с [73] характеристики функциональной безопасности учитывая только те реализации опасных отказов, которые привели к крупному или катастрофическому ущербу в то время, как показатели надежности фиксируют все реализации опасных отказов.

Шубинский И.Б. [122, 123] в своих работах определяет функциональную надежность как готовность системы к выполнению предусмотренных задач. *J.C. Laprier* определяет надежность (*dependability*) как способность предоставлять услуги, которым можно по праву доверять [181, 182].

Согласно ГОСТ 33707-2016 (ISO/IEC 2382:2015) [53], надежность (программного средства): набор атрибутов, относящихся к способности программного обеспечения сохранять свой уровень качества функционирования при установленных условиях за установленный период времени.

В исследованиях Платунова А.Е. [94, 95], Быковского С.В., Горбачева Я.Г., Ключева А.О., Пенского А.В. [33] и других авторов подчеркивается, что

современные технологии разработки нуждаются в усовершенствовании с целью создания эффективных механизмов защиты систем обработки данных и управления, как от злонамеренных воздействий, так и от непреднамеренных ошибок разработчиков на разных стадиях жизненного цикла.

Вопросы обеспечения функциональной надежности в условиях решения задач управления в реальном времени обсуждаются в работах Мостового А.Я. [88], Кириллова Н.П. [63], Муравьевой-Витковской Л.А. [88], *Edward A. Lee* [183].

Результаты анализа научных публикаций указывают на принципиальные различия между подходами к управлению умышленно внесенными дефектами и дефектами, возникшими вследствие ошибок разработчиков. Важно отметить, что непреднамеренные дефекты способны вызывать значительно более серьезные негативные последствия по сравнению с результатами от злонамеренных действий.

Широкое распространение получили методы, ориентированные на выявление ошибок по результатам специально организованных испытаний и изучения исторических данных об опыте эксплуатации ИС. Указанные методы, главным образом, соотносятся с реактивным подходом. Основные цели исследований – выявить закономерности в проявлении дефектов и установить причины возникновения. Данные аспекты находят отражение в работах В. Аджиева [2], *Jeffrey C. Carver* [141], *V. Anu* [130, 131], *Ritu Soni* [206], *Stan Kaplan*, *S. Visnepolschi, B. Zlotin, A. Zusman* [191, 213].

В работах *R. Duphily* [150], *B. Andersen*, A. [129] рассмотрены методы, которые можно обозначить понятием *Root Cause Analysis*. Данная группа методов представляет собой адаптацию методологии *I-TRIZ* в область управления функциональной надежностью.

Методы и модели анализа распространения ошибок (англ. *Error Propagation Analysis*) представлены в работах *Anna Perini* [199], А. Морозова [188, 189], *K. Janschek* [188, 189], *E. Lee* [184]. Суть данной группы методов заключается в раннем обнаружении дефектов и отказов и парировании этих явлений.

*Human Error Taxonomy (HET)* – группа методов ориентированных на предотвращение дефектов, обусловленных ошибками субъективной природы,

представляющих собой отдельную категорию исследований, представленных в работах *Fugun Huang*, [162], *Scott A. Shappell* [203], *J. Reason* [200, 201].

На сегодняшний день вопрос управления дефектами, возникающими из-за субъективных ошибок правообладателей и разработчиков на предпроектной стадии, остаётся малоизученным.

В рамках проактивного подхода к управлению сложными системами развиваются методы, ориентированные на сравнительный анализ альтернатив проектных решений по критериям функциональной надежности (англ. *Failure Mode and Effect Analysis* и *Fault Tree Analysis*). Данные методы нашли отражение в работах *Zhu Y.M.* [216], *Haapanen Pentti* [195], *I. Verzola* [214] и др. исследователей.

В исследованиях Райкова А.Н. [97, 98], Виттиха В.А. [35 - 39], Смирнова С.В. [37] отмечается, что одним из ключевых факторов, снижающих эффективность реализации проектов создания сложных систем, является недостаточное понимание ценностей и ориентиров потенциальных пользователей. Это связано, в частности, с трудностями формирования консолидированного мнения различных целевых групп пользователей и размытости целей функционирования сложных систем, к числу которых относятся ИС.

Проведённый анализ научных публикаций свидетельствует о преобладании исследований, ориентированных на реактивный подход к управлению состоянием сложных систем. Использование только реактивного подхода не позволяет обеспечить требуемый уровень функциональной надежности ИС.

В целом, на сегодняшний день, повышение функциональной надежности ИС достигается преимущественно за счет развития технологий проектирования, тогда как теоретическим аспектам урегулирования проблемных ситуаций, возникающих в ходе реализации проектов ИС, уделяется недостаточное внимание.

**Объект исследования** – функциональная надежность ИС.

**Предмет исследования** – методологическое, методическое, модельное и инструментальное обеспечение функциональной надежности ИС на предпроектной стадии.

**Целью диссертационной работы** является решение актуальной научно-практической проблемы, заключающейся в создании методологических основ обеспечения функциональной надежности информационных систем на предпроектной стадии в виде комплекса принципов, подходов, методов, системных, структурных и математико-статистических моделей и реализованных на их основе инструментальных средств.

### **Задачи исследования**

1. Формирование методологии обеспечения функциональной надежности ИС на предпроектной стадии на основе принципов и подходов системного анализа.
2. Разработка системных моделей предпроектной стадии на основе положений эвергетики.
3. Разработка на основе системных архетипов комплекса структурных, контурных и динамических моделей проблемных ситуаций, связанных с обеспечением функциональной надежности ИС на предпроектной стадии.
4. Разработка методов обеспечения функциональной надежности ИС на предпроектной стадии.
5. Разработка математико-статистических моделей для определения базовых характеристик проекта ИС на основе исторических данных, а также оценка согласованности мнений неоднородных акторов на предпроектной стадии.
6. Разработка на основе новых научных результатов методик и инструментальных средств обеспечения функциональной надежности ИС.

### **Методы исследования**

При проведении исследований использованы методы системного анализа, положения эвергетики, методы контурного и структурного моделирования, математико-статистические методы обработки данных, аппарат теории графов, технологии объектно-ориентированного моделирования и разработки программного обеспечения, отечественные и зарубежные стандарты и руководства, регламентирующие способы решения задач обеспечения функциональной надежности.

## **Положения, выносимые на защиту**

1. Методология обеспечения функциональной надежности ИС на предпроектной стадии на основе принципов и подходов системного анализа.
2. Совокупность системных моделей предпроектной стадии, разработанных на основе положений эвергетики.
3. Комплекс структурных, контурных и динамических моделей проблемных ситуаций, связанных с обеспечением функциональной надежности ИС, на предпроектной стадии на основе системных архетипов, модель возникновения ошибок на предпроектной стадии, модель структуры интегрального показателя, характеризующего совокупное влияние факторов, как способствующих, так и препятствующих повышению функциональной надежности.
4. Комплекс методов обеспечения функциональной надежности ИС на предпроектной стадии, включая: метод построения многомерных зависимостей; метод оценки состояния функциональной надежности информационных сервисов с учетом субъективных оценок пользователей; метод структурного моделирования последствий отказов функциональных компонент ИС.
5. Математико-статистическая модель для определения базовых характеристик проекта ИС на основе исторических данных, а также оценка согласованности мнений неоднородных акторов на предпроектной стадии.
6. Результаты решения практических задач обеспечения функциональной надежности на основе использования полученных научных результатов, включающие: выявление факторов функциональной надежности ИС при формировании единого информационного пространства компании, методику формальной оценки возможности реализации проекта; методику и прототип программного обеспечения для анализа моделей проектируемых компонентов ИС по критериям функциональной надежности.

## **Научная новизна результатов**

1. Научная новизна предложенной методологии обеспечения функциональной надежности ИС на предпроектной стадии заключается в обоснованном формировании и интеграции общенаучных и специальных

принципов и подходов, которые создают методологическую основу для исследования структуры проблемных ситуаций, связанных с обеспечением функциональной надежности ИС на предпроектной стадии. Методология позволяет изучить структуру и разработать открытую совокупность моделей обеспечения функциональной надежности ИС, объединяющей в себе реактивный и проактивный подходы, на предпроектной стадии. (*n.16 паспорта специальности*)

2. Научная новизна предложенных системных моделей заключается в системной сочетании положений эвергетики, *V*-модели жизненного цикла и подхода *QFD* (функции развертывания качества), реализованного в виде модели «Дом качества» - *HoQ*, применительно к задачам обеспечения функциональной надежности ИС на предпроектной стадии. Комплексное использование системных моделей позволяет повысить прозрачность содержания ожидаемых результатов проекта для заинтересованных сторон, что создает основу для выработки неоднородными акторами консолидированного мнения о подходах к реализации ИС. (*n.2 паспорта специальности*)

3. Научная новизна структурных моделей проблемных ситуаций, связанных с обеспечением функциональной надежности ИС на предпроектной стадии, заключается в новом системном сочетании взаимодополняющих моделей архитектур соответствующих разным стратегиям разработки ИС, знаково-ориентированных графов и системных архетипов, что обеспечивает поддержку деятельности, ориентированную на обеспечение требуемого уровня функциональной надежности ИС. Новизна контурных и динамических моделей проблемных ситуаций, связанных с обеспечением функциональной надежности ИС, заключается в представлении системных архетипов совокупностью контурных моделей, весовые характеристики которых определяются математико-статистическими методами, что позволяет перейти от качественного описания ситуаций к математическим моделям, позволяющим получить количественные оценки (статические и динамические) отдельных ситуаций. (*n.n 1, 3 паспорта специальности*)

4. Научная новизна разработанных методов обеспечения функциональной надежности ИС заключается в следующем:

- предложенный метод построения многомерных зависимостей, позволяет представить ИС в виде многосвязного объекта совокупностью одномерных непараметрических регрессионных зависимостей. Метод позволяет исследовать поведение многосвязных объектов в случае, когда некоторые компоненты являются метрическими характеристиками, а некоторые представлены экспертными оценками. (*n.13 паспорта специальности*)

- предложенный метод оценки функциональной надежности информационных сервисов учитывает совокупность субъективных оценок разных пользователей, что позволяет обоснованно определить приоритетное направление повышения функциональной надежности сервисов. (*n.11 паспорта специальности*)

- предложенный метод структурного моделирования последствий отказов функциональных компонент ИС на основе совместного использования аппарата схем сопряжения и таблиц истинности формирует базис для построения структурных моделей отказов в рамках как нисходящего, так и восходящего подходов. Метод позволяет на предпроектной стадии провести оценку возможных последствий отказов функциональных компонент ИС по различным критериям функциональной надежности. (*n.11 паспорта специальности*)

5. Научная новизна предлагаемой модели проекта как многосвязного объекта управления заключается в одновременном использовании как исторических данных о ранее реализованных проектах, представленных в виде числовых характеристик, так и субъективных оценок акторов, причастных к реализации проектов, сопоставимость персонифицированного опыта акторов, накопленного при реализации прежних проектов, выражается посредством функциональных зависимостей. Предложенная формальная модель повышает обоснованность стратегических решений по организации проекта основными заинтересованными сторонами. Новизна предложенного подхода к оценке согласованности мнений неоднородных акторов на основе их субъективного восприятия заключается: во-

первых, в формализации оценки значимости проблемных ситуаций неоднородными акторами. Проблемные ситуации представляются ограниченным набором системных архетипов, а их значимость определяется посредством ранжирования. Во-вторых, в использовании системной модели «Дом качества - HoQ» для формирования на основе совокупности ранговых оценок разных акторов симметричной матрицы парных коэффициентов корреляции. Это, в свою очередь, позволило оценить взаимовлияние влияние мнений акторов посредством парциальных и множественных коэффициентов корреляции.

### **Теоретическая значимость**

Теоретическая значимость диссертационного исследования заключается в развитии теоретических основ, включая методические и модельные основы обеспечения функциональной надежности ИС в условиях разного видения потребительских свойств ИС разными заинтересованными сторонами, что создает основу для совершенствования системы управления функциональной надежностью ИС за счет уменьшения неопределенности состояния и прогнозирования последствий проблемных ситуаций, которые могут возникнуть на предпроектной стадии.

### **Практическая значимость**

На основе предложенных принципов и подходов к обеспечению функциональной надежности выявлены факторы функциональной надежности ИС при формировании единого информационного пространства компаний.

На основе полученных новых теоретических результатов разработана методика и программное обеспечение для оценки альтернативных вариантов реализации ИС, позволяющее оценить на предпроектной стадии предполагаемую удовлетворенность заказчика результатами проекта, а исполнителей – ходом реализации проекта в зависимости от бюджета и ограничений на длительность реализации проекта.

На основе предложенного метода структурного моделирования последствий отказов на основе аппарата схем сопряжения и таблиц истинности разработана методика и программное обеспечение для анализа моделей проектируемых

компонентов ИС по критериям функциональной надежности, позволяющее построить структурные модели отказов функциональных компонент ИС на предпроектной стадии, обоснованно распределить ресурсы на обеспечение функциональной надежности, избавиться от трудоемкого и затратного по времени процесса ручного построения структурных моделей отказов.

Результаты работы внедрены и использованы в «Лукойл - Центрнефтепродукт», Научно-производственном предприятии «Полигон», Инжиниринговой компании «НПФ ЭИТЭК», в компании «Газпромнефть – Цифровые решения», а также в учебном процессе ФГБОУ ВО «Уфимский университет науки и технологий».

### **Степень достоверности и апробация результатов**

Достоверность полученных научных результатов исследования подтверждается корректностью применения общепринятых научных подходов и положений, а также практической апробацией новых научных результатов, подтвержденной соответствующими актами внедрения.

Основные положения и результаты диссертационной работы регулярно докладывались и обсуждались на конференциях, наиболее значимые из которых: Международная конференция «Проблемы управления и моделирования в сложных системах» (ПУМСС), Россия, Самара (2014, 2016, 2018, 2019); Международная конференция “Computer Science and Information Technology” (CSIT’2019, Austria); Международная конференция “Information systems and technologies: achievements and perspectives”, Azerbaijan Republic, Sumgait, 2020; Мультиконференция по проблемам управления, Россия, Санкт-Петербург, 2020; Международная конференция «Электротехнические комплексы и системы» (ICOECS 2020, ICOECS 2021), Россия, Уфа; Международная научно-техническая конференция «Электротехнические комплексы и системы» Россия, Магнитогорск (UralCon 2023), Международная научная конференция «Информационные технологии интеллектуальной поддержки принятия решений» Россия, Уфа (ITIDS’2015, 2018, 2020, 2021, 2024) и др.

## **Связь исследований с научными программами**

Исследования выполнены в рамках грантов РФФИ №19-08-00177 «Методологические, теоретические и модельные основы управления функциональной безопасностью аппаратно-программных комплексов в составе распределенных сложных технических систем», №19-08-00937 «Методы и модели интеллектуальной поддержки принятия решений при управлении программными проектами, реализуемыми в среде производственных предприятий», в рамках основной части государственного задания ВУЗам № FEUE-2023-0007.

## **Публикации**

По теме диссертации опубликовано 40 основных научных работ, в том числе 16 статей в рецензируемых печатных изданиях из перечня, утвержденного ВАК (10 изданий - К1, из них 8 изданий, входит в перечень RSCI, 2 статьи опубликованы автором единолично, 6 изданий – К2); 5 свидетельств о государственной регистрации программы для ЭВМ, 3 монографии, 6 публикаций в международных журналах и изданиях, индексируемых базами WoS и SCOPUS, 16 работ в прочих изданиях.

## **Личный вклад автора**

Результаты исследования, представленные в диссертации, получены лично автором под руководством научного консультанта. Из публикаций, подготовленных в соавторстве, в диссертации представлен материал, принадлежащий лично автору.

**Структура и объем работы.** Диссертационная работа состоит из введения, шести глав, заключения, списка литературы из 216 наименований, изложенных на 318 страницах машинописного текста, содержит 165 рисунков и 36 таблиц. Объем приложений составляет 25 страниц.

## **Благодарности**

Автор выражает глубокую благодарность и признательность научному консультанту, профессору, доктору технических наук В. Е. Гвоздеву за огромный труд в проводимых в течение многих лет исследованиях, а также за всестороннюю помощь и поддержку при подготовке диссертации к защите.

# ГЛАВА 1. АНАЛИЗ ПРОБЛЕМНОЙ СИТУАЦИИ ОБЕСПЕЧЕНИЯ ФУНКЦИОНАЛЬНОЙ НАДЕЖНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ НА ПРЕДПРОЕКТНОЙ СТАДИИ

Данная глава посвящена анализу проблемной ситуации обеспечения функциональной надежности ИС на предпроектной стадии.

Первый параграф настоящей главы посвящен вопросам актуальности проблемы обеспечения функциональной надежности ИС. Приводятся трактовки понятия «функциональная надежность» различными учеными. Рассматриваются предпосылки возникновения проблемных ситуаций, связанных с обеспечением функциональной надежности ИС.

Второй параграф посвящен вопросам эволюции подходов к обеспечению функциональной надежности ИС, затрагиваются вопросы изменения масштабов и сложности систем.

В третьем параграфе рассмотрены основные подходы к обеспечению функциональной надежности ИС, включающие активный, реактивный и проактивный подходы. Приводится анализ современного состояния исследований в области управления функциональной надежностью, установлено, что большинство методов преимущественно ориентированы на реализацию реактивного подхода.

В четвертом параграфе сформулирована цель исследования, определен круг научных и практических задач, базовые положения исследований, сформирована основная научная идея работы: комплексное исследование проблемы обеспечения функциональной надежности ИС на предпроектной стадии на основе положений эвергетики.

## 1.1 Актуальность проблемы обеспечения функциональной надежности информационных систем

В современных условиях определяющим направлением построения ИС выступает концепция сетевентрического управления [80]. Система сетевентрического управления представляет собой сочетание заранее созданных и развернутых разветвленных автоматизированных электронных сетей сбора и обработки первичной информации, узлов хранения и анализа информации, а также контуров управления и принятия решений [68, 69]. Их интеграция формирует единое информационное и управляемое пространство (англ. *System of Systems-SoS* [175, 209]).

Ключевым свойством таких систем является интероперабельность, обеспечивающая возможность совместного функционирования и обмена данными между разнородными компонентами, подсистемами и участниками в рамках распределённой информационной среды.

Интероперабельность (англ. *interoperability*) - ключевое свойство сетевентрических систем, обеспечивающее их способность к совместному функционированию и обмену данными между разнородными компонентами, подсистемами и участниками в рамках распределённой информационной среды [65, 79, 128]. Интероперабельность остаётся фундаментальным требованием для эффективного функционирования сетевентрических систем, особенно в условиях роста сложности и масштаба цифровых экосистем.

Особенностью формирования единого информационного пространства является переход от централизованного управления требованиями к функциональным характеристикам ИС к децентрализованным, что предполагает выработку заинтересованными сторонами консолидированных решений на предпроектной стадии.

Ключевая роль единого информационного пространства в задачах сетевентрического управления обуславливают важность задачи обеспечения

функциональной надежности локальных ИС, системная интеграция которых формирует единое информационное пространство.

Под функциональной надежностью ИС понимается свойство сохранять работоспособность в соответствии со своим целевым назначением при случайных дестабилизирующих воздействиях и отсутствии злоумышленного влияния на программную и аппаратную составляющую.

В условиях постиндустриального общества [58], когда единое информационное пространство становится критическим фактором сетецентрического управления, содержание понятия «работоспособность» зависит от ситуации, для урегулирования которой субъектам, вовлеченным в урегулирование, необходима информационная поддержка, соответствующая их информационным потребностям.

Понятие «функциональная надежность» и «функциональная безопасность» до настоящего времени трактуется различными исследователями неоднозначно.

В рамках стандарта МЭК 61508-2-2012 [47], функциональная безопасность (англ. *functional safety*) определяется как корректное функционирование системы управления и управляемого оборудования. Липаев В.В. [73, 74, 76] в своих работах считает, что понятия и характеристики функциональной безопасности систем близки к понятиям надежности. Основное различие согласно [73, 101], состоит в том, что в показателях функциональной безопасности учитывают только те реализации опасных отказов, которые привели к крупному или катастрофическому ущербу, а в показателях надежности учитываются все реализации опасных отказов.

Согласно ГОСТ 33707-2016 (ISO/IEC 2382:2015) [53], надежность (программного средства): набор атрибутов, относящихся к способности программного обеспечения сохранять свой уровень качества функционирования при установленных условиях за установленный период времени.

Шубинский И.Б. [122] в своих работах определяет функциональную надежность как готовность системы к выполнению предусмотренных задач.

Согласно J.C. Laprier [181, 182] надежность (англ. *Dependability*) - способность предоставлять услуги, которым можно по праву доверять (англ. *Dependability of a system is the ability to deliver service that can justifiably be trusted*).

На рисунке 1.1 представлено дерево функциональной надежности, согласно J.C. Laprier и Шубинскому И.Б.

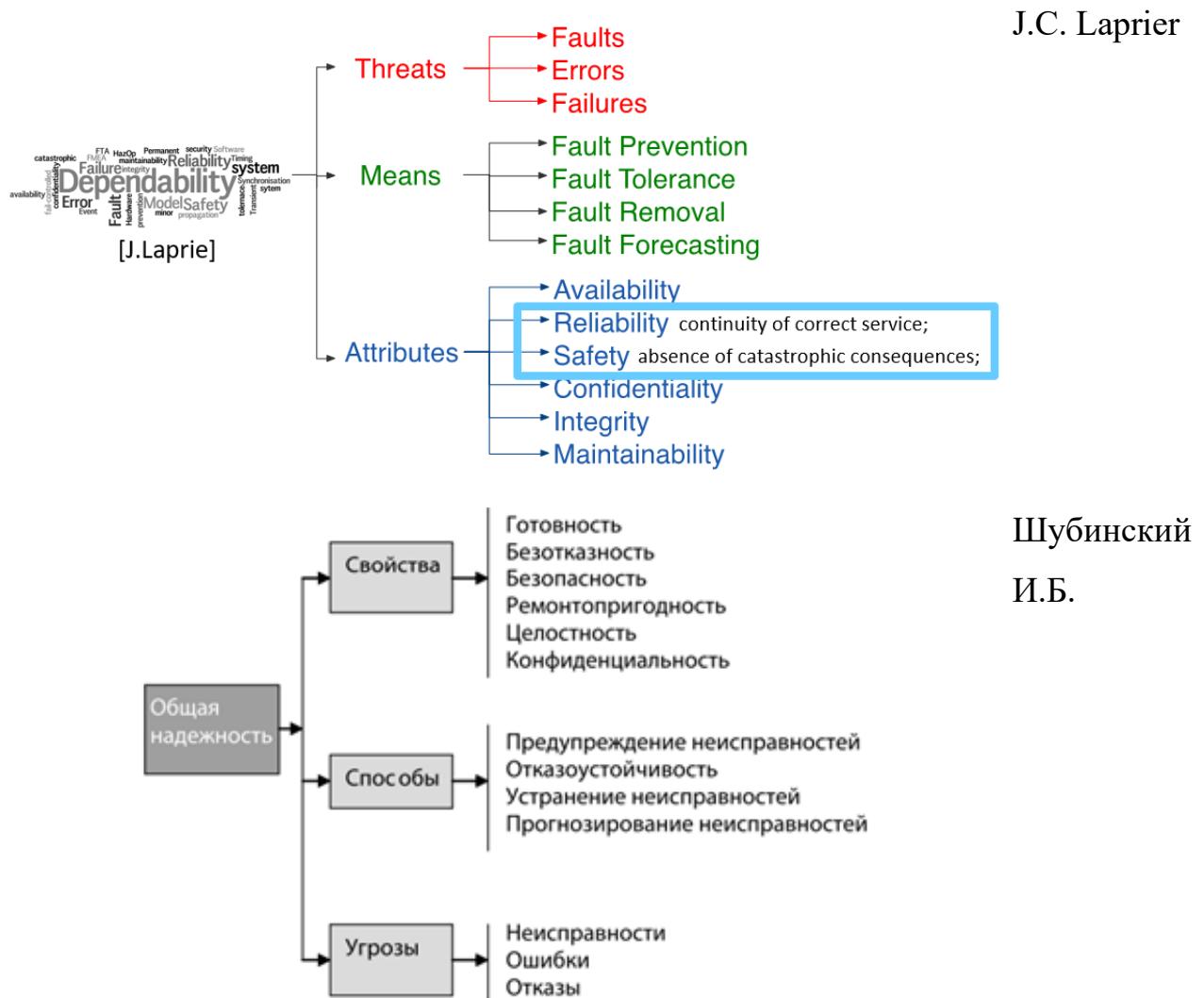


Рисунок 1.1 – Дерево надежности согласно J.C. Laprier и Шубинскому И.Б.

«Общая надежность» согласно [122, 123] представляет собой интегральное понятие и охватывает следующие свойства или атрибуты:

- готовность для правильного обслуживания;
- безотказность, т.е непрерывность правильного обслуживания;
- безопасность – отсутствие катастрофических последствий для пользователя и окружающей среды;
- конфиденциальность – отсутствие неуполномоченного раскрытия информации;
- целостность – отсутствие ненадлежащего изменения состояния системы;

– ремонтопригодность – способность системы к ремонту и обновлениям.

В данной работе под функциональной надежностью ИС понимается свойство сохранять работоспособность в соответствии со своим целевым назначением при случайных дестабилизирующих воздействиях и отсутствии злоумышленного влияния на программную и аппаратную составляющую.

Функциональная надежность определяется способностью ИС своевременно предоставлять пользователю полную информацию. Основным понятием надежности является отказ (англ. *failure*). Для сложных систем определение этого понятия осложняется оценкой последствий отказа отдельных компонентов для свойств системы в целом [77, 101, 176].

Функциональная надежность является латентной характеристикой ИС и объективно характеризуется удовлетворенностью потребителей способностью системы в нужное время предоставлять сведения, соответствующие ситуации, которую нужно урегулировать. Отмечается, что потребительские свойства (включая функциональную надежность) ИС определяются качеством управления проектами разработки ИС [152].

Субъектоцентрическая природа проекта разработки ИС предопределяет то, что организационные, методические и технологические ошибки, допускаемые субъектами проектирования, являются неизменной составляющей любого проекта.

Это подчёркивается во многих литературных источниках, например в аналитических отчетах *Standish Group* [143 - 145].

По методике *Standish Group* ИТ-проект считается успешным, если уложился в согласованные заказчиком и исполнителем срок и не перерасходовал первоначальный бюджет и в нем были реализованы согласованные с заказчиком требования к характеристикам и функциям системы. Результаты двадцатилетних исследований *Standish Group* показали, что доля успешных ИТ-проектов в среднем составляет 30%, а доля неуспешных и проблемных – 70% [4, 143 - 145].

Необходимо подчеркнуть, что несмотря на развитие методических и инструментальных основ проектирования эти соотношения сохраняются в силу увеличения масштабов и сложности ИС, а также неопределенности условий использования.

По данным *Standish Group* свыше 50 % программных проектов имеют частичный или полный провал (англ. *fail*). К такому заключению пришли на основании анализа результатов работы более 50 000 компаний. По результатам отчетов, также это подчеркивается в [13, 149], можно выделить основные факторы, отрицательно влияющие на ход программного проекта:

- неполнота и неоднозначность требований, а также их частая корректировка в ходе реализации проекта;
- завышенные ожидания и нечеткие цели;
- низкое качество управления проектом, в том числе обусловленное неопределенностью среды реализации;
- технологическая некомпетентность (низкая квалификация);
- недостаток ресурсов и нереалистичные временные границы.

На современном этапе повышение надежности ИС в первую очередь связывается с совершенствованием технологий проектирования [75]. При этом вопросам разработки теоретических подходов к урегулированию проблемных ситуаций, неизбежно возникающих в процессе управления проектами разработки ИС, уделяется значительно меньше внимания.

Различная динамика возникновения разных сочетаний проблемных ситуаций обуславливает разные подходы к их урегулированию, что находит выражение в содержании и динамике изменения функциональных и нефункциональных требований к информационному обеспечению деятельности, связанной с урегулированием проблемных ситуаций.

В совокупности уникальность задач обеспечения функциональной надежности ИС определяется следующими аспектами:

- различие содержания, масштаба и сложности проблемных ситуаций;
- множественность факторов, являющихся причиной проблемных ситуаций и их сложное сочетание;
- слабая предсказуемость возникновения и динамики проблемных ситуаций, особенности дестабилизирующих факторов, влияющих на функциональную надежность.

Таким образом, изменение масштабов и сложности ИС, формирующих единое информационное пространство (англ. *System of Systems - SoS*) не позволяет покрыть все множество объективно существующих задач обеспечения функциональной надежности ИС посредством ранее разработанных принципов, подходов, моделей и методик.

Понимание среды разработки и использования системы/SoS является опорной точкой выбора наилучшего способа реализации принципов системной инженерии [46, 209] в имеющей место конкретной ситуации.

Также требуется совершенствование теоретических основ управления проектами с учетом не только разнообразия платформ, положенных в основу разработки функциональных компонент ИС, но и особенностей управления их состоянием, обеспечивающих бесшовную интеграцию не только на технологическом, но и на семантическом и организационном уровнях. Отмеченное обстоятельство обуславливает актуальность задачи совершенствования теоретической, методической, модельной и инструментальной базы обеспечения функциональной надежности ИС на предпроектной стадии.

## **1.2 Эволюция подходов к обеспечению функциональной надежности информационных систем**

Одним из основополагающих аспектов системного анализа является принцип историзма, суть которого заключается в изучении объектов исследования в динамике, с учётом их изменений во времени и пространстве. Анализ эволюции подходов к обеспечению функциональной надежности ИС позволяет выделить факторы, влияющие на изменение парадигм обеспечения функциональной надежности.

Проследить изменение масштабов и сложности систем можно на примере концепции Индустрия 4.0 [99, 164, 1653] (рисунок 1.2).

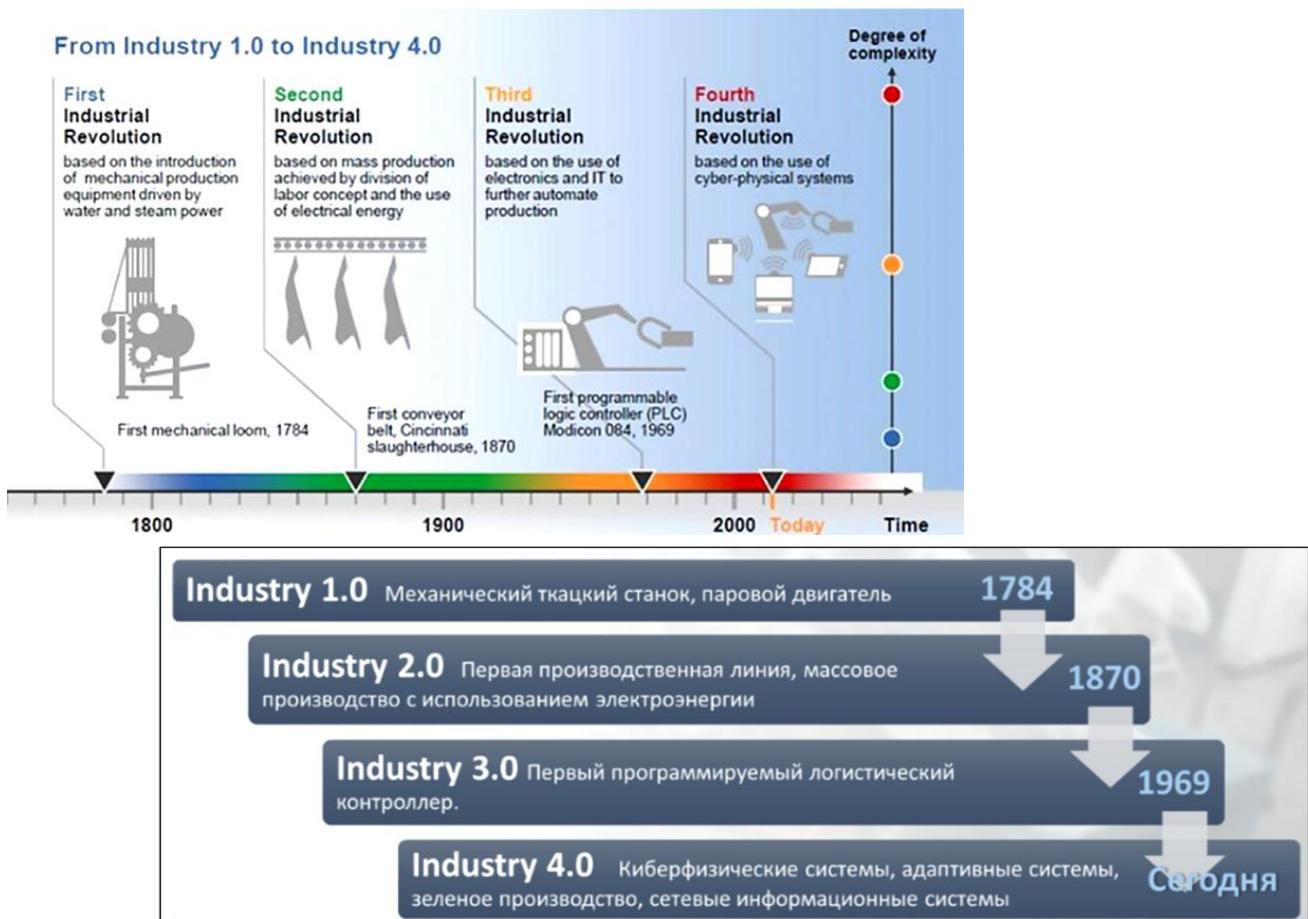


Рисунок 1.2 – Изменение масштабов и сложности систем<sup>1</sup>

Концепция Industry 4.0 возникла в Германии в начале 2010-х годов как стратегическая программа модернизации промышленности. Основное внимание уделяется автоматизации, киберфизическими системам и «умным фабрикам».

В России как национальный приоритет с 2017 года действует программа «Цифровая экономика Российской Федерации». В программе подчеркивается роль данных как стратегического ресурса, развитие цифровых платформ, электронной торговли, финтеха и онлайн-сервисов. Ключевая задача - не только цифровизация отдельных отраслей, но и формирование единого цифрового (информационного) пространства страны.

Таким образом, можно говорить о наличии общих тенденций: акцент на сквозных цифровых технологиях, формирование единого информационного пространства и усиление роли данных как основы для принятия управленческих решений.

<sup>1</sup> Источник: Siemens, Picture of the Future, Spring 2013

Формирование единого информационного пространства выдвигает на первый план задачу обеспечения функциональной надёжности ИС.

В процессе эволюции подходов к решению задачи обеспечения функциональной надежности ИС выделяют несколько основных этапов:

- первые подходы (1950-е годы);

В этот период обеспечение функциональной надёжности было основано на четких постановках вычислительных задач, высокой степени формализации алгоритмов вычислений и подходов к тестированию. Отладка осуществлялась путём поиска и исправления ошибок в коде и аппаратуре. Разработка программного обеспечения была основана на интуиции и опыте разработчиков. Отсутствовали чёткие стандарты и методологии разработки.

В силу того, что постановщик задач, разработчик и пользователь часто являлись одним субъектом, вопросы формализации и обеспечения качества требований не получили достаточного развития.

- период структурного программирования (1960 – 1970-е годы);

Для данного периода характерно появление первых методологий разработки, таких как структурное программирование. Использование структурного подхода к проектированию и разработке программ, предполагало разбиение программы на отдельные модули, каждый из которых выполнял свою функцию. Это позволило упростить тестирование и отладку программ, а также повысить их надёжность, а также обеспечить модернизацию и сопровождение программных продуктов.

- модульное тестирование (с середины 1960-х годов);

Модульное тестирование предполагает проверку каждого модуля программы отдельно от других модулей. Для этого используются специальные тестовые наборы данных, которые позволяют проверить правильность работы модуля в различных ситуациях. Для данного периода характерно использование статистических методов для анализа и контроля качества программного обеспечения, внедрение стандартов и процедур обеспечения качеств.

- автоматизированное тестирование (с середины 1980-х годов);

Автоматизированное тестирование позволяет ускорить процесс тестирования и снизить вероятность ошибок, связанных с человеческим фактором. Существуют различные инструменты для автоматизированного тестирования, такие как модульные тесты, интеграционные тесты, функциональные тесты и т. д.

- объектно-ориентированный подход (с 1990-х годов по настоящее время);

Развитие объектно-ориентированного подхода к разработке программного обеспечения повысило внимание к тестированию и обеспечению качества на всех этапах разработки.

- методология разработки *Agile* (с 2001г. по настоящее время);

Развитие данной методологии стало ответом на сложность и непредсказуемость проектов по разработке программного обеспечения. В соответствии с данной методологией проект разбивается на короткие циклы, что позволяет быстро реагировать на изменения требований и условий. *Agile* предполагает активное взаимодействие между членами команды, а также с заказчиком. Постоянная обратная связь от заказчика и членов команды позволяет выявлять и устранять ошибки на ранних этапах разработки.

Хотя к данному периоду уже появились стандарты по управлению требованиями (например, *ESA-PSS-05-02*, *ESA-PSS-05-03*), однако не получило достаточного развития управление качеством предпроектного обследования и формирования технического задания.

Также следует отметить, что внедрение *Agile* требует определённой культуры и навыков внутри организации, важно понимать, что *Agile* не является универсальным решением для всех проектов и требует адаптации под конкретные условия и требования.

В настоящее время обеспечение функциональной надёжности ИС является сложной задачей, требующей комплексного подхода и использования современных методов и технологий. Одним из базовых положений обеспечения функциональной надежности ИС является постоянное отслеживание соответствия потребительских свойств целям и ценностям пользователей, что, в частности предполагает постоянное отслеживание степени удовлетворенности пользователей надежностью ИС [113, 145]. Полученные результаты служат основанием

обеспечения коэволюции потребительских свойств ИС изменяющимся в пространстве и во времени целям и представлениям о ценностях пользователей.

Программная инженерия длительное время акцентировала основное внимание на развитии теоретической базы и инструментальных средств поэтапного преобразования технического задания в ИС. С одной стороны, функциональная надежность ИС напрямую связана с надежностью аппаратной составляющей. В данной области осталось незначительное количество нерешённых задач, электроника демонстрирует стабильную работу на протяжении многих лет без отказов, а если этого недостаточно, то всегда есть возможность резервирования. Функциональная надежность программных компонентов, имеющих не меньшее значение, является недостаточной, на сегодняшний день описано множество случаев [2, 210], когда ошибки в ПО обходились в миллионы долларов<sup>2</sup> или даже приводили к гибели людей<sup>3</sup>.

Недостаточное качество программных компонентов, в том числе, обусловлено уровнем развития теоретической и методической базы учета субъектоцентрической природы ИС на предпроектной стадии.

Субъектоцентрическая природа проекта предопределяет то, что организационные, методические и технологические ошибки, допускаемые субъектами проектирования, являются неизменной составляющей любого проекта. Ошибки эти обусловлены совокупностью факторов: различием представлений о ценностях ключевых правообладателей; сложностью формирования субъектами проекта консолидированной онтологической модели, являющейся платформой для выработки согласованных решений; неоднородностью и ограниченностью знаний, опыта, глубины понимания текущих и перспективных информационных потребностей, ограниченностью ресурсов проекта.

<sup>2</sup> 4 июня 1996 года случился неудачный запуск ракеты-носителя *Ariane 5*, которая была разработана Европейским космическим агентством. Ракета разрушилась на 39-й секунде полета из-за неверной работы бортового программного обеспечения. Это одна из самых дорогостоящих компьютерных ошибок

<sup>3</sup> В результате авиакатастрофы с *Boeing 737 Max* в Эфиопии, произошедшей в марте 2019г. погибли 157 человек. Комиссия по расследованию авиакатастрофы, объявила, что на потерпевшем крушение самолете эфиопских авиалиний было дефектное бортовое программное обеспечение, кроме того, авиаконцерн недостаточно подготовил пилотов.

### **1.3 Анализ современного состояния исследований, связанных с обеспечением функциональной надежности информационных систем**

В трудах [139, 201] и многих других обращается внимание на необходимость смещения акцентов в проблемах создания ИС от вопросов штатной эксплуатации к вопросам их безопасного функционирования. В других исследованиях, в частности в [162, 203], акцентируется внимание на необходимости улучшения технологий разработки, которые позволили бы обеспечить защиту систем обработки данных и управления как от злонамеренных действий, так и от непреднамеренных ошибок, допускаемых разработчиками на ранних стадиях жизненного цикла ИС, включая предпроектную стадию.

Возникновение ошибок на предпроектной стадии в значительной степени обусловлено неопределенностью, которая является необъемлемым свойством всех сложных систем. Эта особенность ИС подробно обсуждается, например, в [73].

В научных трудах [33, 40, 76, 91] подчёркивается важность развития методологической и теоретической базы дефектологии ИС как одного из направлений исследований в области системной инженерии.

В результате анализа литературных источников можно сделать вывод о наличии концептуальных различий в подходах к управлению дефектами, которые вносятся в ИС преднамеренно, и дефектами, возникающими случайно из-за ошибок разработчиков ИС.

Необходимо акцентировать внимание на том, что масштабы негативных последствий от непреднамеренных ошибок могут значительно превышать последствия преднамеренных действий, чему в истории существует множество примеров, подтверждающих это (например, [2, 70, 210]).

Используемые методы обеспечения функциональной надежности ИС можно соотнести с проактивным, сценарным (активным) и реактивным подходами к управлению сложными системами (см. рисунок 1.3).

При использовании реактивного подхода (Тестирование) целью является: доказать отсутствие ошибок в ИС.

При использовании сценарного подхода – цель: доказать соответствие потребительских свойств ИС представлениям о ценностях пользователей. Основная цель проактивного подхода – не допустить преобразования латентных дефектов в негативные последствия.

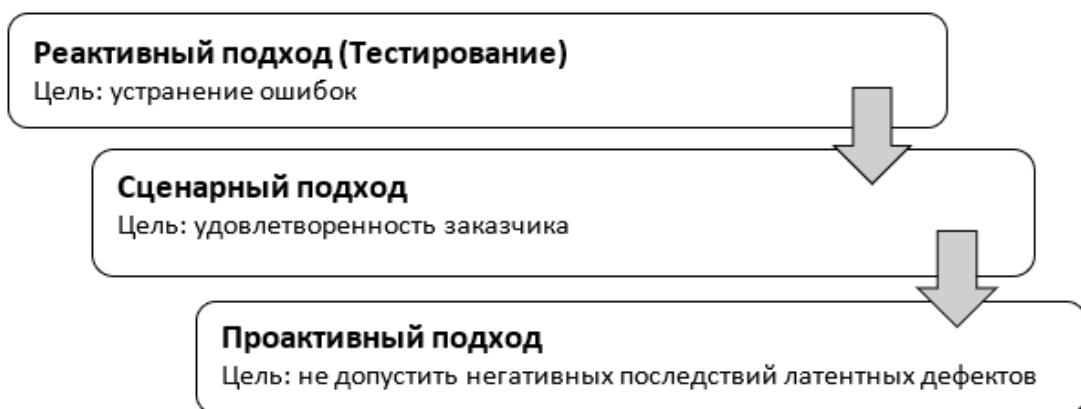


Рисунок 1.3 – Подходы к обеспечению функциональной надежности ИС

В настоящее время преобладают методы, которые соотносятся с активным и реактивным подходами, речь идёт о методах, направленных на выявление ошибок по итогам специально организованных испытаний.

Одним из традиционных методов поиска ошибок является тестирование. *Тестирование* процесс выполнения программы с целью нахождения ошибок [78]. Этот процесс может быть выполнен как вручную, так и с использованием автоматизированных инструментов.

В сфере тестирования традиционно выделяют следующие основные виды:

- модульное тестирование, заключается в тестировании отдельных методов и функций классов, компонентов или модулей, используемых в программном обеспечении;
- интеграционное тестирование оценивает возможность подключения и передачи данных между несколькими модулями или сервисами;

- функциональное тестирование, основное внимание уделяется бизнес-требованиям к программному обеспечению, функциональные тесты проверяют только результат некоторого действия и не проверяют промежуточные состояния системы при выполнении этого действия;
- системное тестирование, также известное как сквозное тестирование сценариев, позволяет оценить пользовательский опыт работы с приложением; т.е. сквозное тестирование копирует поведение пользователя при работе с программным обеспечением;
- приемо-сдаточное тестирование проводится для проверки соответствия системы требованиям заказчика.

Примеры видов тестирования показаны на рисунке 1.4. Методы тестирования включают в себя порядка 150 различных подходов, и их количество продолжает расти, что обусловлено, во-первых, ростом требований к функциональной надежности, во-вторых, ростом масштаба и сложности систем, формирующих единое информационное пространство сетевоцентрического управления.



Рисунок 1.4 – Виды тестирования

Известны и активно используются в настоящее время методы изучения исторических данных об опыте эксплуатации ИС. Цель указанных методов - установление закономерностей в проявлениях симптомов дефектов, а также причин возникновения дефектов [150, 202, 206, 213].

Вопросы обеспечения функциональной надежности и безопасности ИС в контексте решения задач управления в режиме реального времени активно обсуждаются в [63, 87, 88, 83, 95], в работе [90] предложены архитектурные решения для минимизации рисков в критически важных системах.

*Root Cause Analysis (RCA)* - группа методов, представляющих собой адаптацию принципов методологии *I-TRIZ* [3, 129, 146, 159].

*RCA* представляет собой совокупность технологий, которые представляют собой систематический, документируемый, основанный на количественных показателях подход к выявлению, пониманию, объяснению и устраниению коренных причин наблюдаемых проблем.

Целью *RCA* является выработка и согласование действий, реализация которых позволит исключить, либо хотя бы уменьшить влияние упомянутых причин, негативно влияющих на свойства сложных систем разной природы, что является основой их долговременных улучшений [3, 129, 147, 206, 25].

*RCA* – инструмент определения истинных причин произошедшего, а также способов обеспечения того, чтобы это не повторилось вновь. Отвечает на вопросы, что произошло? и почему произошло? как исключить возможность повторения произошедшего? Проведение *RCA* предполагает использование разного, либо нескольких инструментов для выделения истинной причины (в дальнейшем именуемые «коренными причинами») и отделения ее от симптомов и маскирующих факторов.

Для *RCA* не создано единой методологии. В основе подхода – совокупность инструментов, процессов, философий. *RCA* предполагает реализацию последовательности следующих шагов, представленных на рисунке 1.5.



- 1) определение проблемы;
- 2) сбор данных;
- 3) выявление причинно-следственных факторов;
- 4) выявление коренных причин;
- 5) разработка рекомендаций по устранению коренных причин;
- 6) реализация рекомендаций по устранению коренных причин.

Рисунок 1.5 – Последовательность шагов *RCA*

Эффективное применение метода анализа коренных причин (*RCA*) предполагает использование разнообразных инструментов.

– Диаграмма Исикиавы (*Fishbone*, «Рыбья кость»).

Инструмент для анализа проблемы, который способствует разноаспектному комплексному изучению факторов, выявляя первопричины и формируя представление о проблемной ситуации. Также известен как метод Исикиавы или диаграмма Исикиавы.

Диаграмма представляет собой графическое изображение, которое помогает анализировать и определять причинно-следственные связи между проблемой и факторами, влияющими на неё, пример показан на рисунке 1.6.

Диаграмма представляет визуальный инструмент для выявления первопричин проблем путем классификации потенциальных причин по различным категориям. Такой подход позволяет структурировать рассуждения и обеспечивает единое понимание исследуемой ситуации.

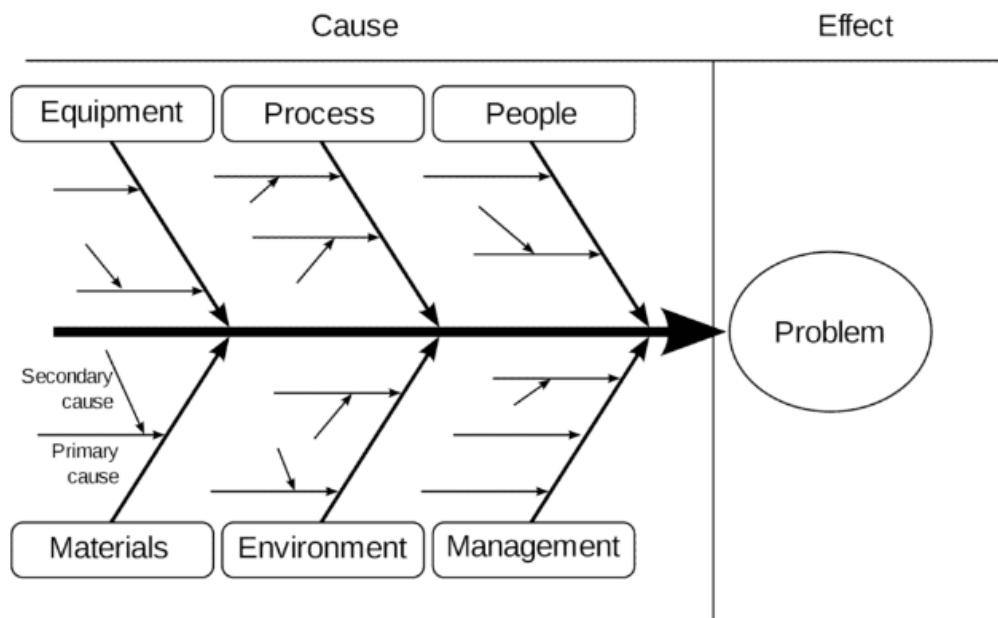


Рисунок 1.6 – Диаграмма Исикиавы (*Fishbone*)

Данный метод эффективен для совместной работы в команде, улучшения понимания сложных проблем и систематического подхода к их решению. В контексте ИС диаграмма Исикиавы помогает выявить факторы, влияющие на функциональную надежность. Однако метод имеет существенные ограничения, такие как потенциальная сложность при работе с большими наборами данных, субъективность участников анализа способна повлиять на интерпретацию причин, а также диаграмма не предполагает количественной оценки.

- Метод анализа первопричин «5 почему» (5 Why?) (рисунок 1.7);

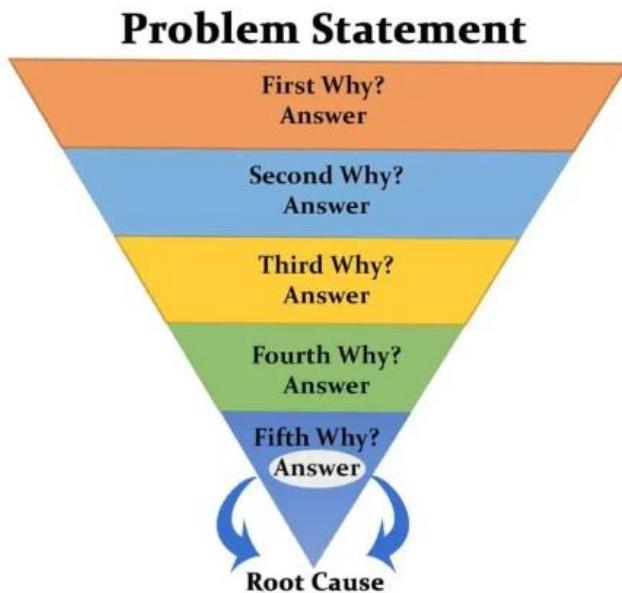


Рисунок 1.7 – Метод анализа первопричин «5 почему?»

Метод постановки вопросов, который предполагает последовательное применение вопроса «почему?» (*Why?*) до тех пор, пока не будет найдена фундаментальная причина проблемы.

Данный инструмент обеспечивает глубокое погружение в структуру причинно-следственных связей, что способствует тщательному анализу исследуемых вопросов. Итеративный характер фокусирует внимание не на непосредственных причинах недостаточного качества объектов управления, а на выявлении основных факторов, устранение которых повлечет за собой исчезновение непосредственных причин.

Эффективность методологии обусловлена её простотой и понятностью, что делает её полезным инструментом для оперативного устранения проблем и выявления первопричин, однако ограничением методологии является субъективность и отсутствие формальных моделей. В условиях работы со сложными системами метод анализа первопричин «5 почему» часто используется в сочетании с другими инструментами анализа — (например, с диаграммой Исикавы или деревом отказов, которые позволяют визуализировать и систематизировать выявленные зависимости). В этом случае методика вопросов служит первым шагом, помогающим выделить ключевые направления для последующего анализа.

Однако ограничением метода является субъективность и отсутствие формальных моделей.

– Анализ Парето.

Данный инструмент, основанный на Принципе Парето, способствует определению приоритетов в проблемах путём концентрации внимания на ограниченном количестве значимых аспектов вместо рассмотрения широкого круга тривиальных вопросов (пример показан на рисунке 1.8).

Данный принцип предполагает, что примерно 80% последствий происходят от 20% причин. В ИС анализ Парето используется для выявления источников наиболее часто встречающихся ошибок. Из работ *J. Reason* [200, 201] возможно предположить, что в ИС примерно 80 % ошибок происходят от 20% латентных дефектов.

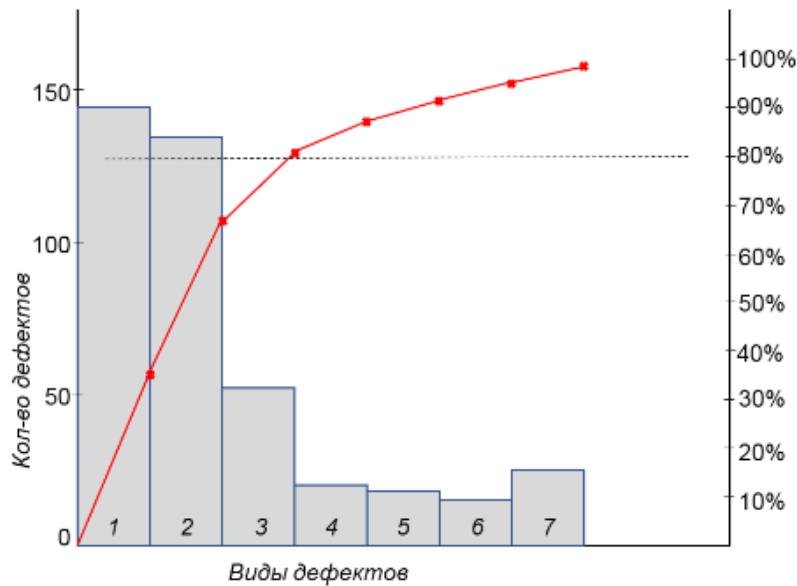


Рисунок 1.8 – Анализ Парето (пример)

В контексте *RCA* это означает выявление ключевых факторов, которые способствуют возникновению проблемы. Данный инструмент реализует эффективный подход к *RCA*, полезен в условиях ограниченности ресурсов. Однако данный инструмент не объясняет взаимосвязь между причинами, что требует использования его в комплексе с другими инструментами *RCA*.

В работах [178, 191, 204] представлен метод, относящийся к направлению *Anticipatory Failure Determination (AFD)*, известный в отечественной научной среде под наименованием «диверсионный анализ». *AFD* представляет собой обобщение известных подходов сценарного анализа [177, 211, 214, 193]. Концептуальной базой диверсионного анализа (*AFD*) является идея о том, что процессы выявления и идентификации отказов носят творческий характер, но при этом могут быть реализованы в рамках системного подхода.

Несмотря на высокую ценность, *AFD* имеет ограничения: результаты во многом зависят от опыта и креативности экспертов, участвующих в построении сценариев. С точки зрения анализа функциональной надежности ИС подход *AFD* имеет следующие ограничения – ввиду отсутствия ограничений: во-первых, он не позволяет рассматривать временной аспект, а во-вторых, в его рамках не учитывается роль субъекта в субъектоцентрических системах.

В рамках активного подхода к обеспечению функциональной надёжности ИС выделяют направление, включающее методы и модели анализа распространения ошибок, (*Error Propagation Analysis*) [82, 148, 199, 188 – 190]. Основная идея этих методов заключается в раннем выявлении проявлений дефектов и отказов, а также парирование этих явлений.

Следует выделить группу методов, учитывающих субъектоцентрическую природу ИС и ориентированных на предотвращение дефектов, обусловленных ошибками субъективной природы [151, 158, 203, 205].

Согласно определению, приведенному в [200], ошибки обусловлены недостатком когнитивной деятельности (*Human Error*), которая связана с разработкой подходов к урегулированию проблемной ситуации, разработке плана реализации подхода, реализации плана. Эти ошибки в дальнейшем проявляются в виде дефектов в спецификациях требований. Выявление проблемных ситуаций, в которых возникают недостатки в когнитивной деятельности, создают предпосылки для создания барьеров, которые могли бы препятствовать возникновению ошибок. Наибольшее распространение получили таксономия ошибок в требованиях – *Requirements Error Taxonomy (RET)* [130, 160, 180] и таксономия ошибок людей – *Human Error Taxonomy (HET)* [131, 141, 197, 200].

Основу *RET* составляет перечень ошибок, допускаемых при разработке требований. Наличие такого перечня исключает полную зависимость результатов анализа дефектов от креативности инспектора требований. Таксономия ошибок содержит три основные категории:

- ошибки людей (*people errors*);
- ошибки процессов (*process errors*);
- ошибки документирования (*documentation errors*).

Таксономия *RET* фокусируется на работе с ошибками, проявляющимися в форме организационных дефектов при управлении программными проектами.

В результате экспериментов было установлено, что *RET* позволяет существенно увеличить эффективность реализации программных проектов, а также то, что треть выявленных дефектов обусловлена ошибками, которые относятся к первой категории – ошибок людей.

В когнитивной психологии известно несколько систем классификации ошибок, наиболее известной является классификация *J. Reason HET* [198], основу которой составляет обобщенная модель обработки информации людьми. Данная модель включает три этапа:

- 1) ощущения и восприятие;
- 2) обработка информации и принятие решения;
- 3) реализация действий.

Ошибки могут возникать на любой из вышеописанных стадий. *J. Reason* разделил ошибки, совершаемые людьми, на три класса на основе их сходства:

- *slips* (например, канцелярские ошибки, небрежности в документировании, несогласованность содержания требований в спецификаций);
- *lapses* (упущенные из внимания сведения, полученные от правообладателей, случайно упущенные требования);
- *mistakes* (к числу наиболее распространенных когнитивных и организационных ошибок относятся: некорректная семантическая интерпретация, необоснованные гипотезы, ошибочная оценка состояния окружающей среды, неверное понимание функциональных ролей в организационной структуре, ошибки принятия решений ошибки принятия решений и др.).

*Slips* и *lapses* имеют место при совершении действий, которые были запланированы, и являются следствием невнимательности и рассеянности.

*Mistakes* являются следствием ошибок на стадии планирования, когда выбираются действия, несоответствующие ситуации.

*J. Reason* устанавливает следующую связь между преобразованием информации и выделенными классами ошибок, представлено на рисунке 1.9.

Ограничением *HET* является то обстоятельство, что таксономия применима только к случайно совершающим ошибкам.

Такие ошибки, как предположение о возможности реализовать проект на основе неполных требований или игнорирование некоторых задач, относящихся к инженерии требований, т.е. преднамеренные нарушения, не относятся ни к одному из классов *HET*.

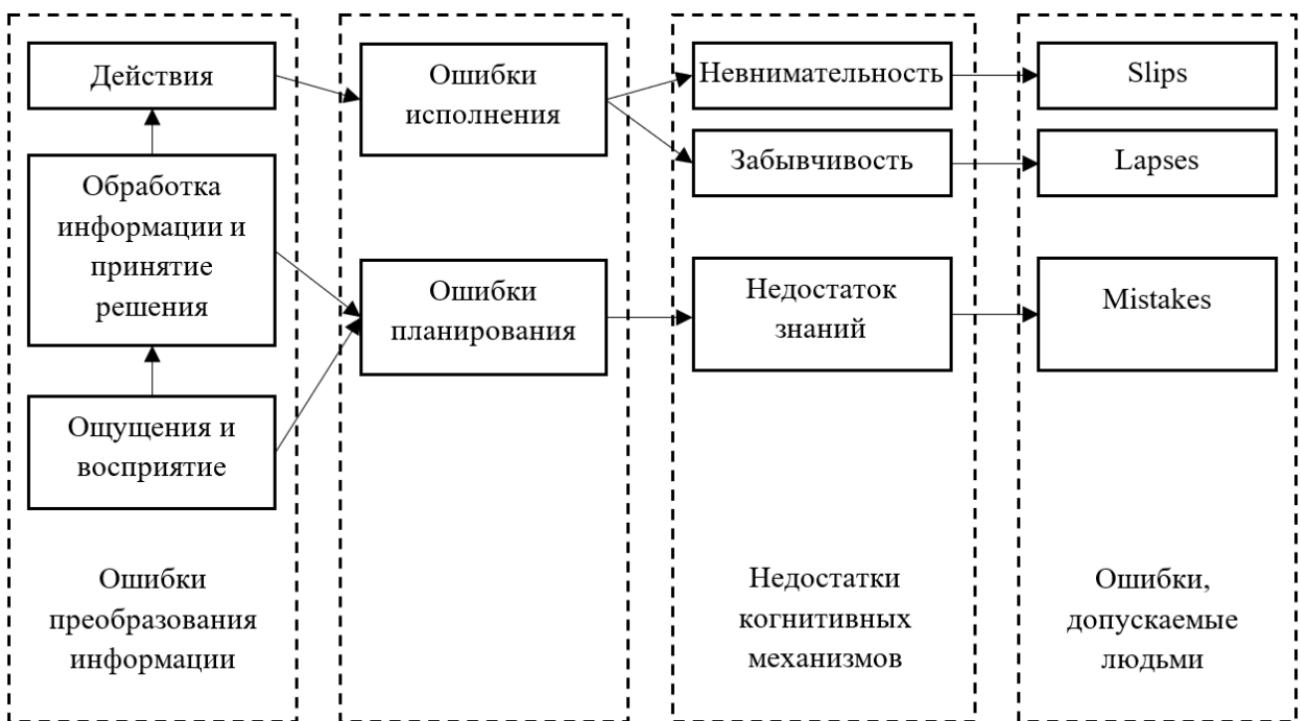


Рисунок 1.9 – Механизмы возникновения ошибок по J. Reason

Перспективным направлением исследований в области системной инженерии является обеспечение функциональной надежности субъектоцентрических систем (к классу которых относятся ИС) с использованием подхода «барьерного мышления» (англ. – *barrier thinking*).

Данное направление основывается на метафоре «Модель швейцарского сыра» – *Swiss Cheese Model (SCM)* [200, 201], авторами которой являются *J. Wreathall и J. Reason*. Концепция возникла на основе анализа инцидентов, в ходе которого было установлено, что их причиной зачастую становятся непреднамеренные латентные дефекты. Эти дефекты обусловлены ошибками и действиями людей, удаленными от инцидента в пространстве и во времени.

На рисунке 1.10. схематично представлена «Модель швейцарского сыра».

Инцидент рассматривается как результат совмещения на одной линии «дыр», при существующих компонентах (слоях) многослойной системы. «Дыры» рассматриваются как динамически возникающие объекты. Причиной возникновения «дыр» является активизация случайными внешними воздействиями латентных дефектов, присутствующих в слоях системы.

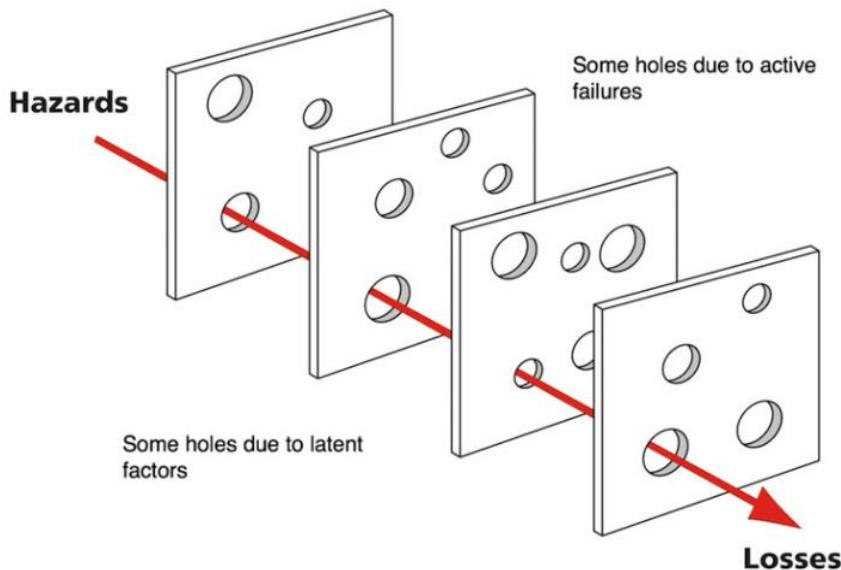


Рисунок 1.10 – «Модель швейцарского сыра» – «*Swiss Cheese Model*»

Ключевой тезис барьерного мышления заключается в признании того, что для любого объекта управления неизбежно наличие источников опасности для объекта управления, а также латентных дефектов в самом объекте управления.

Основная цель подхода состоит в предотвращении комбинации таких латентных дефектов на разных уровнях управления системой. Это достигается путём формирования защитных барьеров, не допускающих перехода потенциальных опасностей в нежелательные события.

В зависимости от направленности исследований, которые определяют содержание метафоры *SCM*, исторически различают три вида системных моделей инцидентов:

- модель *Mark-I*;
- модель *Mark-II*;
- модель *Mark-III*.

Модель *Mark-I*. Согласно [201], концептуальную основу построения модели составили выделенные *J. Reason* и *J. Wreathall* пять базовых проекций объекта управления:

- принятия субъективных решений на высшем уровне управления (англ. *top level decision makers*);

- линейного (тактического) управления (англ. *line management*);
- выявление предпосылок к возникновению инцидентов (англ. *preconditions*);
- производственной деятельности (англ. *productivity activity*);
- парирования инцидентов (англ. *defences*).

Построенная многослойная модель использует выделенные проекции в качестве основы. В слоях модели «дыры» (*holes*) соответствуют событиям/условиям, результатом размещения которых на одной линии является инцидент.

В модели *Mark-I* выделены следующие слои:

- психологических предпосылок (*psychological precursors*);
- опасных действий (*unsafe acts*);
- эшелонированной системы защиты (*defence-in-depth*).

Схематично модель представлена на рисунке 1.11.

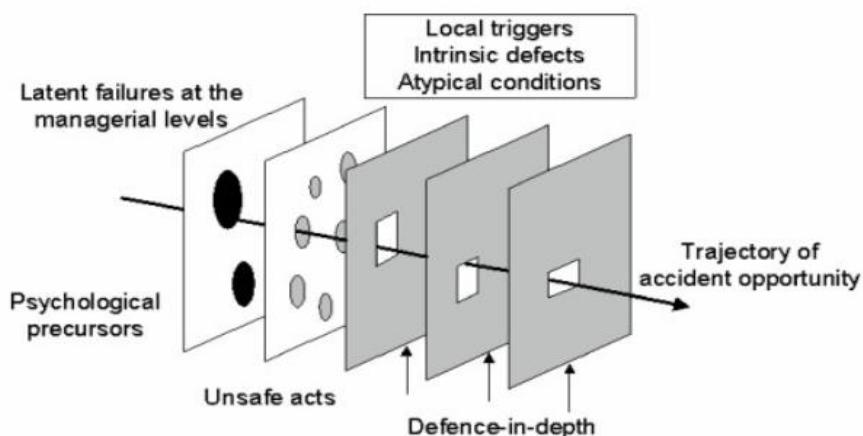


Рисунок 1.11 – Модель *Mark-I*<sup>4</sup>

Активные внешние воздействия (*local triggers*), внутренние дефекты (*intrinsic defects*) и нарушение штатных условий функционирования систем (*atypical condition*) приводят к возникновению «дыр» в слоях «опасные действия»

---

<sup>4</sup> источник Eurocontrol Experimental Centre. Revisiting the «Swiss Cheese» Model of Accidents, EEC Note No. 13/06 Project Safbuild, 2006

и «эшелонированная система защиты». Если эти дыры оказываются на одной линии с латентными дефектами слоя «психологические предпосылки», возникают предпосылки к возникновению инцидента.

Каузальные цепочки берут начало на верхнем (организационном) уровне и заканчиваются инцидентами. Выявление подобных цепочек позволяет объяснить причины инцидентов, в том числе обусловленные непредсказуемым стечением обстоятельств. Слой «психологические предпосылки» подчеркивает то обстоятельство, что ошибки высшего руководства при назначении полномочий и зон ответственности исполнителем (т.е. организационные, системные ошибки) в том числе являются причинами инцидентов.

Модель *Mark-II*. Согласно [201], в модели *Mark-II* водится понятие «барьер». Инциденты объясняются не только неблагоприятным стечением обстоятельств, но и отказами в защитных барьерах. Главным вопросом последствия инцидентов становится не «кто виноват?», а «почему не сработала система защиты».

Смещение акцентов от источников опасности к защитным барьерам выражает следующую точку зрения: невозможность заранее предвидеть, в какой среде будет функционировать система, ограничивает эффективность подходов, ориентированных исключительно на выявление источников опасности. Более обоснованным подходом является усиление защитных свойств системы и разработка барьерных механизмов, чем выявление внешних источников негативных воздействий.

Важным отличием *Mark-II* от *Mark-I* является то, что к числу базовых компонентов верхнего (организационного) уровня отнесена корпоративная культура, характеризующая отношение к ошибкам, совершаемым людьми, и подходы к организации процессов. Признаком высокой корпоративной культуры является различное отношение к ошибкам, обусловленным обстоятельствами непреодолимой силы (например, фундаментальной неопределенностью проблемной ситуации; возникновением условий, вынуждающих идти на нарушение правил).

Модель схематично представлена на рисунке 1.12.

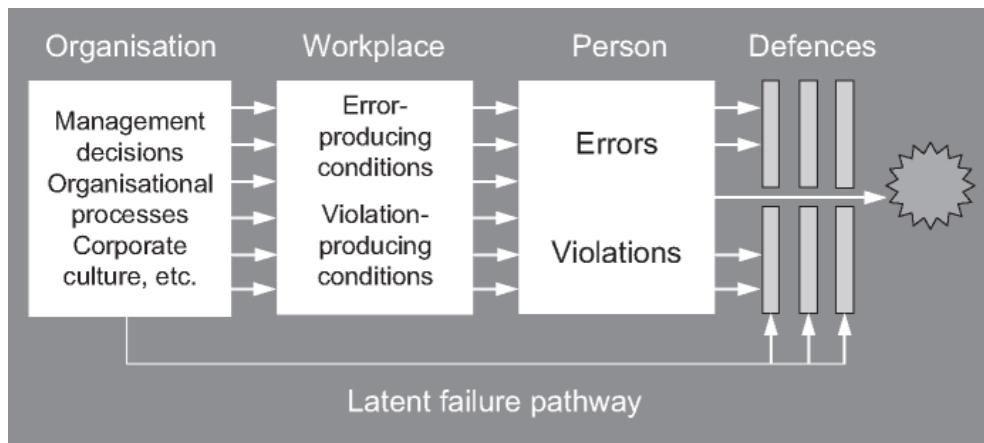


Рисунок 1.12 – Модель Mark-II<sup>5</sup>

Модель Mark-III. Согласно [201], фокусом исследований становится не инциденты, а связанные с ними негативные последствия (потери ценностей) для субъектов. Инцидент возникает в результате непредсказуемого возникновения каузальной цепочки событий, обусловленных сочетанием внешних событий и латентных дефектов. Инцидент может явиться результатом сочетания событий, вероятность одновременного возникновения каждого из которых крайне мала.

В рамках *Mark-III* подчеркивается, что исследование инцидентов нельзя сводить лишь к выявлению каузальных цепочек. Выявление «коренных причин» не позволяет сформировать достаточную основу для предупреждения дефектов.

В методическом плане особенностью *Mark-III* (представлено на рисунке 1.13) является положение о том, что в состав модели помимо прочих обязательно должны входить три концепта: источник опасности; система защиты; ущерб. Количество «слоев сыра» заранее не регламентируется и определяется исходя из особенностей исследуемой системы. Каждый из слоев имеет барьеры, препятствующие возникновению инцидента.

К трудно определяемым причинам, создающим условия для возникновения латентных дефектов, относятся те причины, которые обусловлены неверными организационными решениями высшего руководства, неверными проектными и конструкторскими решениями, дефектами в технической и эксплуатационной документации.

<sup>5</sup> источник Eurocontrol Experimental Centre. Revisiting the «Swiss Cheese» Model of Accidents, EEC Note No. 13/06 Project Safbuild, 2006

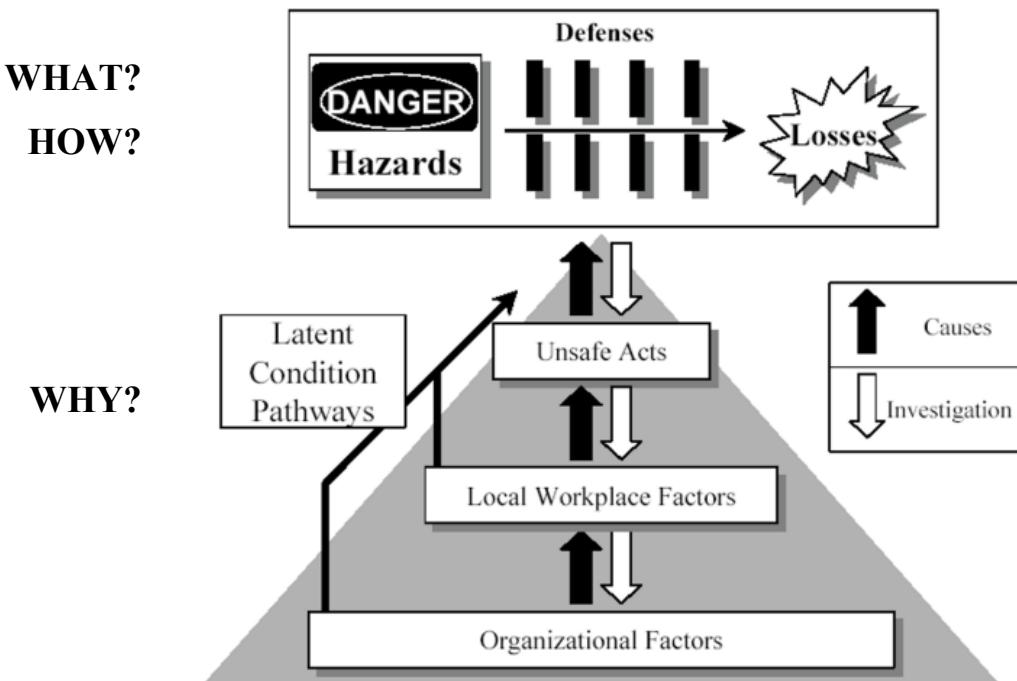


Рисунок 1.13 – Модель *Mark-III*<sup>6</sup>

Таким образом, *SCM* представляет собой теоретическую основу, используемую для интерпретации причин инцидентов, имеющих место в хорошо защищенных системах. Основу концепции составляет положение о том, что никакой инцидент не может быть обусловлен единственной причиной (например, отказом техники или ошибкой персонала).

Помимо создания основы для объяснения и анализа, *SCM* ориентирует на выделение ограниченного набора показателей, исследование изменений которых во времени создает предпосылки для решения прогностических задач [198].

Развитием метафоры *SCM* в область открытых систем является «модель галстук-бабочки – *Model Bow Tie (MBT)*» [137].

Построение прикладных моделей в рамках *MBT* в соответствии с [137] предполагает реализацию последовательности шагов:

- идентификация опасностей (*hazard*); выявление «поворотных точек» (*top events*), т.е. таких состояний объекта управления, при которых опасность реализовалась, но еще есть возможность предотвратить потери;

<sup>6</sup> источник Eurocontrol Experimental Centre. Revisiting the «Swiss Cheese» Model of Accidents, EEC Note No. 13/06 Project Safbuild, 2006

- выявление факторов, приводящих к возникновению «поворотных точек» (*threat*); оценка последствий (*consequence*),
- являющихся следствием реализации опасностей; идентификация защитных барьеров, предупреждающих реализацию опасностей (*threat barriers*);
- идентификация защитных барьеров, препятствующих трансформации опасности в потери в случае её реализации (*recovery preparedness measure*);
- идентификация факторов/условий эскалации, ослабляющих защитные свойства барьеров (*escalation factor*);
- идентификация защитных барьеров, предупреждающих реализацию факторов/условий эскалации (*escalation factor control*).

Основным назначением модели *MBT* является получение качественных оценок о возможных причинах и последствиях инцидентов.

Схематично последовательность шагов в соответствии с [137] показана на рисунке 1.14).

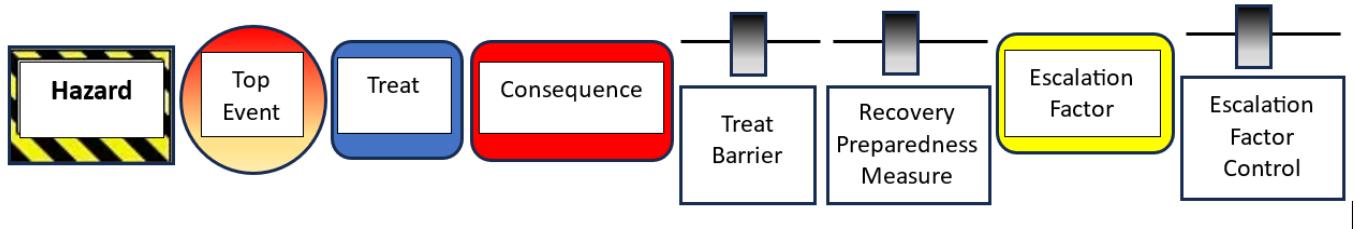


Рисунок 1.14 – Последовательность *MBT* (*Bow Tie analysis in 8 Steps*)

«Модель Галстук-бабочки» (*MBT*) представляет собой диаграмму, которая показывает основные сценарии возникновения опасных событий, а также установленные барьеры, направленные на предотвращение или минимизацию возможных негативных последствий. Схематично, пример *MBT* показан на рисунке 1.15.

Данный метод позволяет наглядно, просто и понятно визуализировать проблему. Однако, существенным ограничением *MBT* является отсутствие возможности учитывать и отображать совокупности причин, которые возникают одновременно и вызывают последствия.

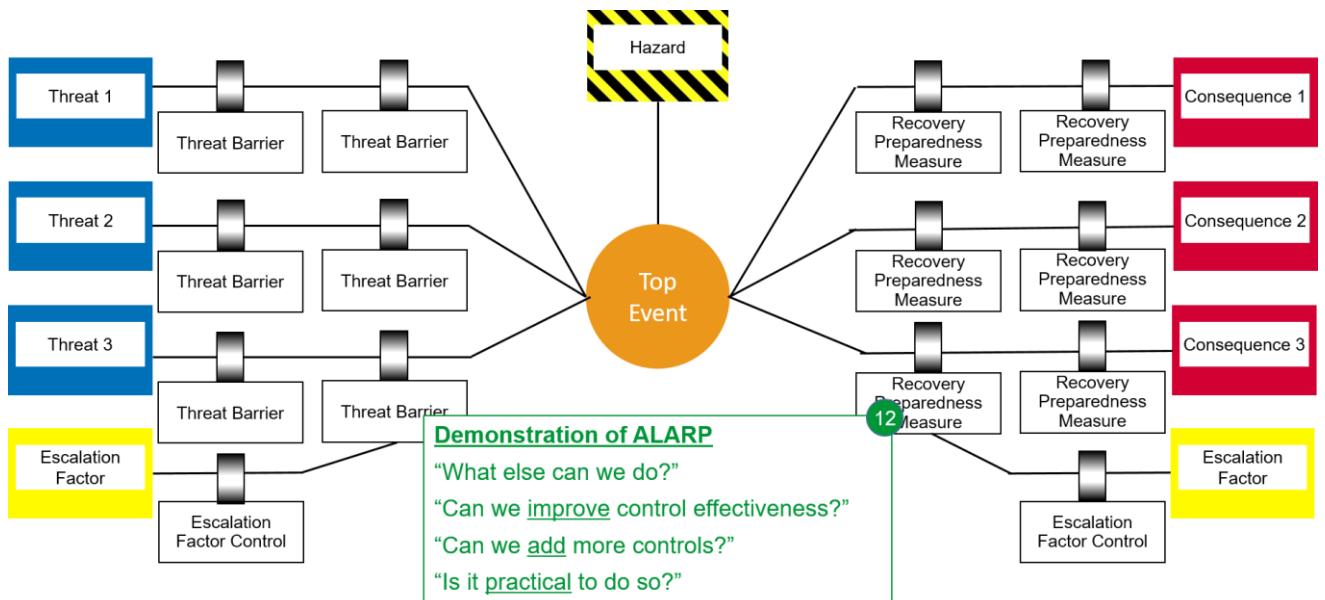


Рисунок 1.15 – Пример MBT

Метод анализа дерева неисправностей [39] (англ. *Fault Tree Analysis (FTA)*) и метод анализа дерева событий [43] (англ. *Event Tree Analysis (ETA)*) можно отнести к проактивному подходу к управлению сложными системами.

Данные методы ориентированы на сравнительный анализ альтернатив проектных решений по критериям функциональной надежности [40].

*Event Tree Analysis (ETA)* – анализ дерева событий. Данный метод – это индуктивный анализ отказов, который выполняется, чтобы определить последствия единичного отказа для надежности всей системы, т.е. реализация восходящего подхода к проектированию [215].

Дерево событий (*Event Tree*) – это визуальная презентация последствий единичного отказа, а также его влияния на другие события и систему в целом. Пример *ETA* показан на рисунке 1.16.

Недостатком метода является то, что анализ ограничен одним исходным событием, также метод не работает для нескольких сценариев, которые включают неопределенность, такую как человеческие ошибки.

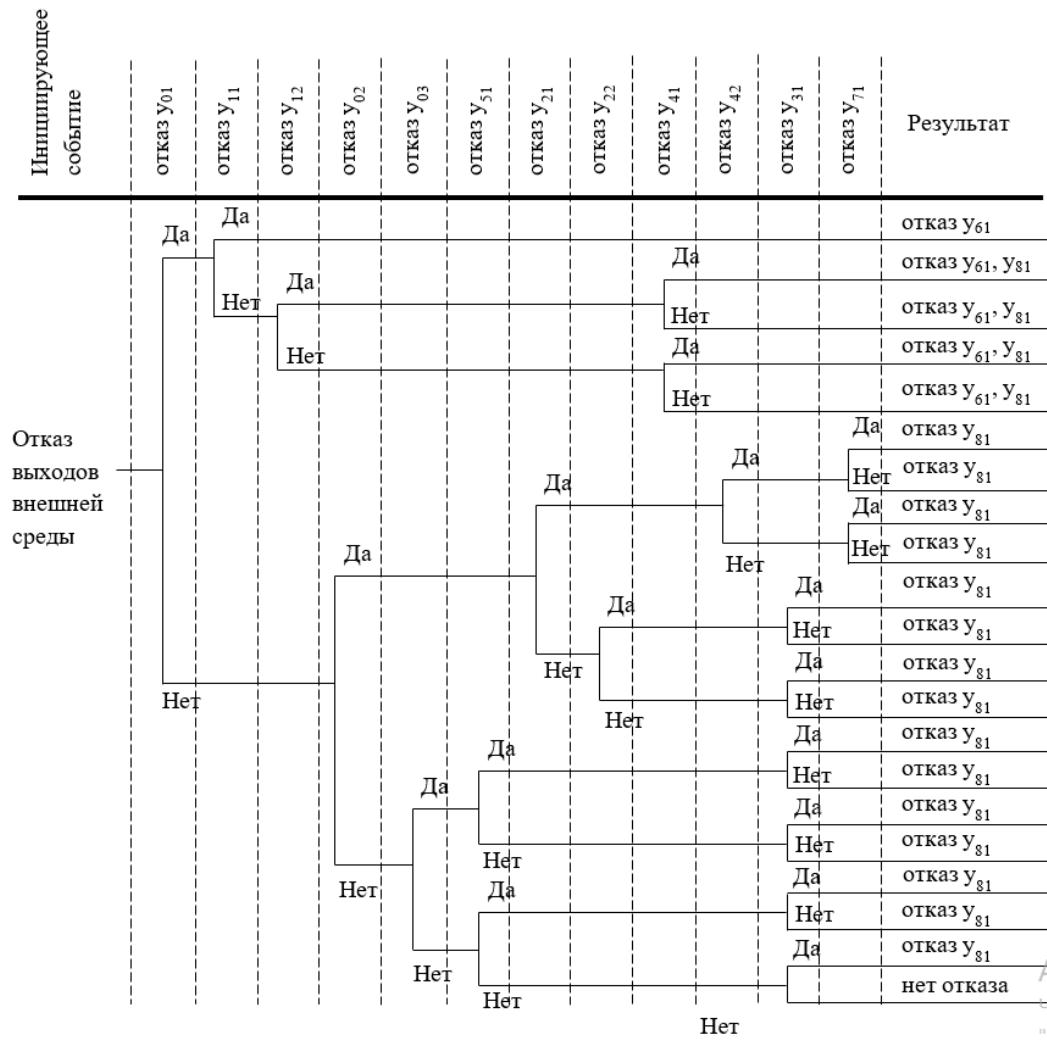


Рисунок 1.16 – Пример ETA

*Fault Tree Analysis (FTA)* – анализ дерева отказов. *FTA* представляет собой упорядоченное графическое представление логико-вероятностной связи случайных событий (отказов, ошибок и т. д.), приводящих к реализации нежелательного конечного события. Логический анализ дерева отказов полезен при идентификации простых путей отказа в сложных системах, где комбинации событий могут привести к возникновению конечного события. *FTA* реализует нисходящий подход к проектированию [184, 195, 196, 211]. Пример *FTA* показан на рисунке 1.17.

На практике часто используют комбинацию *ETA* (рисунок 1.16) и *FTA* (рисунок 1.17), называемую *Causes and Consequences Analysis (CCA)* - анализ причин и последствий.

Ограничением метода является то, что в модели не учитывается временной аспект, дерево отказов является статичной моделью и не позволяет исследовать цепные реакции (эффект домино).

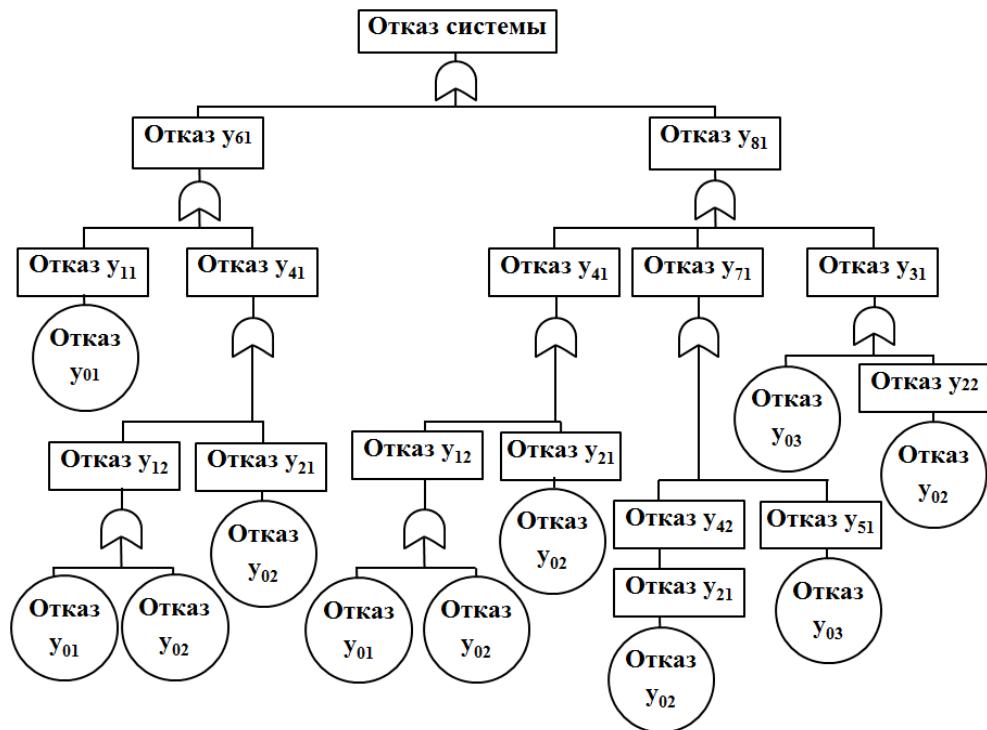


Рисунок 1.17 – Пример FTA

В исследованиях [13, 113, 143 – 145] делается акцент на том, что одним из решающих факторов, препятствующих успешной реализации проектов по разработке ИС, является низкое качество функциональных и нефункциональных требований к потребительским свойствам систем. Это, в том числе, обусловлено сложностью формирования консолидированного мнения ключевых правообладателей в условиях неопределенности среды использования и нечеткой постановки целей функционирования сложных систем [97, 98, 35].

В работах В.А. Виттиха [35 - 39] разрабатывается теория интерсубъективного управления. Предложено понятие «эвергетика» – субъективно-ценностно-ориентированная наука о процессах интерсубъективного управления в сложных системах. Центральным понятием эвергетики является «неоднородный актор» – субъект, вовлеченный в урегулирование проблемной ситуации. Вовлеченность означает заинтересованность субъекта в изменении ситуации и обладание полезными, с точки зрения урегулирования ситуации, ресурсами.

Формирование интерсубъективной системы сводится «... к рождению (на основе самоорганизации) сообщества, представляющего собой объединение акторов (с имеющимися в их распоряжении ресурсами), которые обладают единым пониманием того, какие вопросы в рамках настоящего объединения регулируются совместно, а какие самостоятельно каждым актором...» [35].

В работе [120] с позиций эвергетики рассматриваются вопросы обоснования положения о необходимости выявления дефектов, соотносимых с фазой предподготовки проектирования систем обработки данных. Применительно к проблеме функциональной надежности ИС теория интерсубъективного управления выражается в необходимости развития подходов к выработке консолидированных решений в условиях различия онтологических и когнитивных моделей неоднородных акторов, вовлеченных в проектирование и эксплуатацию ИС.

В таблице 1.1 кратко представлены рассмотренные методы, соотнесенные с реактивным и проактивным подходами, а также основные публикации.

Таблица 1.1 – Краткий обзор методов исследований функциональной надежности

Подходы к управлению сложными системами	Методы	Основные научные публикации
Активный и реактивный	<i>Root Cause Analysis (RCA)</i> Анализ коренных причин	<ul style="list-style-type: none"> <li>– <i>B. Andersen, T. Fagerhaug. Root cause analysis of the. Simplified tools and techniques</i> [127];</li> <li>– <i>Cognitive Approach to Root Cause Analysis for Improvement Quality of Life: A Case Study for IT industry</i> [147];</li> </ul>
	<i>Error Propagation Analysis (EPA)</i> Методы, ориентированные на раннее обнаружение проявлений дефектов и сбоев и парировании этих явлений	<ul style="list-style-type: none"> <li>– <i>A. Morozov, K. Janschek. Dual Graph Error Propagation Model for Mechatronic System Analysis</i> [186];</li> <li>– <i>D. Nassar, W. Abdel Moez and etc. Error Propagation Analysis of Software Architecture Specifications</i> [190];</li> <li>–</li> </ul>
	Тестирование программ	<ul style="list-style-type: none"> <li>– Г. Майерс, Т. Баджетт, К. Сандлер. Искусство тестирования программ [78];</li> <li>– С. С. Куликов. Тестирование программного обеспечения. Базовый курс [71];</li> </ul>

Подходы к управлению сложными системами	Методы	Основные научные публикации
	<i>Anticipatory Failure Determination (AFD-1)</i>  Методы, ориентированные на поиск причины произошедшего сбоя	<ul style="list-style-type: none"> <li>- <i>S. Visnepolschi, B. Zlotin, A. Zusman.</i> New tools for failure and risk analysis: An Introduction to anticipatory failure determination (AFD) and the theory of scenario structuring [213].</li> <li>- <i>G. Klein, D. Snowden, L.P. Chew.</i> Anticipatory Thinking [178].</li> </ul>
Проактивный	<i>Failure Mode and Effect Analysis (FMEA); Fault Tree Analysis (FTA);</i>  Методы, ориентированные на сравнительный анализ альтернатив проектных решений по критериям функциональной надежности	<ul style="list-style-type: none"> <li>- <i>Zhu Y.M.</i> Failure-Modes-Based Software Reading [216];</li> <li>- <i>Silvia Tolo, John Andrews Tolo.</i> Fault Tree Analysis Including Component Dependencies [211];</li> <li>- <i>C. Thurnes, F. Zeihsel, S. Visnepolschi, F. Hallfell.</i> Using TRIZ to invent failures - concept and application to go belong traditional FMEA [212];</li> </ul>
	<i>Anticipatory Failure Determination (AFD-2)</i>  Методы, ориентированные на прогнозирование сбоя	<ul style="list-style-type: none"> <li>- <i>G. Klein, D. Snowden, L.P. Chew.</i> Anticipatory Thinking [178];</li> <li>- <i>R. Silva, M. Carvalho.</i> Anticipatory Failure Determination (AFD) for product reliability analysis: A comparison between AFD and Failure Mode and Effects Analysis (FMEA) for identifying potential failure modes [204];</li> </ul>
	Анализ ошибок людей (HET, RET)  Методы, ориентированные на предотвращение дефектов, обусловленных ошибками субъективной природы.	<ul style="list-style-type: none"> <li>- <i>James S. Reason.</i> Human Error [200];</li> <li>- <i>Paul Grünbacher, Anna Perini (Eds.)</i> Requirements Engineering for adaptive system: Foundation for Software Quality [190];</li> <li>- <i>V. K. Anu.</i> Using Human Error Models to Improve the Quality of Software Requirements [130];</li> </ul>
	<i>Swiss Cheese Model</i>	<ul style="list-style-type: none"> <li>- <i>James S. Reason, Hollnagel E, Paries J.</i> Revising the «Swiss Cheese Model» of Accidents [201];</li> </ul>
	<i>Bow Tie Model</i> (барьерное мышление)	<ul style="list-style-type: none"> <li>- BowTieXP. The next generation BowTie methodology tool. BowTie Methodology Manual Revision 15 [137];</li> <li>- И.С. Жуков. Барьеры безопасности: основные понятия, обзор, концепция [59];</li> </ul>
	Методы, развивающиеся в рамках интерсубъективного управления и эвергетики	<ul style="list-style-type: none"> <li>- Виттих В.А. Неоднородный актор и повседневность как ключевое понятие эвергетики [36].</li> </ul>

Проведенный анализ подходов позволил сделать заключение о том, в исследованиях преобладает реактивный подход. Тем не менее, реактивный подход не способен обеспечить требуемый уровень функциональной надежности ИС. Существенно меньшая часть исследований сфокусирована на управлении дефектами, вызванными субъективными ошибками правообладателей и разработчиков, причастных к реализации проектов ИС. Методы, реализующие проактивный подход, носят, как правило, концептуальный характер.

В основном в современных исследованиях основной акцент сделан на разработку и улучшение методов проектирования, в то же время теоретические аспекты разрешения проблемных ситуаций, которые неизбежно возникают при реализации проектов ИС, получили значительно меньше внимания. Одним из перспективных направлений исследований представляется адаптация аппарата системных архетипов в область управления проектами разработки ИС.

#### **1.4 Цель и задачи исследований**

В связи с тем, что ИС, являются в настоящее время системообразующим фактором сетевентрического управления, проблема функциональной надежности относится к числу приоритетных проблем исследований.

К настоящему времени теория надежности автоматических технических систем получила мощное развитие. Известно достаточно много локальных подходов, методов и моделей, так или иначе, связанных с обеспечением функциональной надежности. Однако изменение масштабов и сложности систем, формирование единого информационного пространства (*System of Systems - SoS*) не позволяет покрыть все множество объективно существующих задач обеспечения функциональной надежности ИС посредством ранее разработанных подходов.

К настоящему времени не сформировалась законченная теория обеспечения функциональной надежности, в полной мере учитывающая субъектоцентрическую природу ИС. Проведенный анализ проблемной ситуации обеспечения функциональной надежности ИС, а также анализ существующих подходов и

методов обеспечения функциональной надежности позволил сформулировать цель исследования и выполнить постановку задач исследования.

Целью настоящей работы является решение актуальной научно-практической проблемы, заключающейся в создании методологических основ обеспечения функциональной надежности информационных систем на предпроектной стадии в виде комплекса принципов, подходов, методов, структурных и математико-статистических моделей и реализованных на их основе инструментальных средств.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Сформировать методологию обеспечения функциональной надежности ИС на предпроектной стадии на основе принципов и подходов системного анализа.

2. Разработать системные модели предпроектной стадии на основе положений эвергетики.

3. Разработать на основе системных архетипов комплекс структурных, контурных и динамических моделей проблемных ситуаций, связанных с обеспечением функциональной надежности ИС на предпроектной стадии.

4. Разработать методы обеспечения функциональной надежности ИС на предпроектной стадии.

5. Разработать математико-статистические модели для определения базовых характеристик проекта ИС на основе исторических данных, а также выполнить оценку согласованности мнений неоднородных акторов на предпроектной стадии.

6. Разработать на основе новых научных результатов методики и инструментальные средства обеспечения функциональной надежности ИС.

Стратегия исследований, включающая совокупность решаемых задач, базовые положения и взаимосвязи между основными направлениями системного анализа, представлена на рисунке 1.18. Представленная стратегия отражает логико-структурную организацию работы. Основная научная идея, лежащая в основе диссертационного исследования, заключается в комплексном исследовании проблемы обеспечения функциональной надежности ИС на предпроектной стадии на основе положений эвергетики.

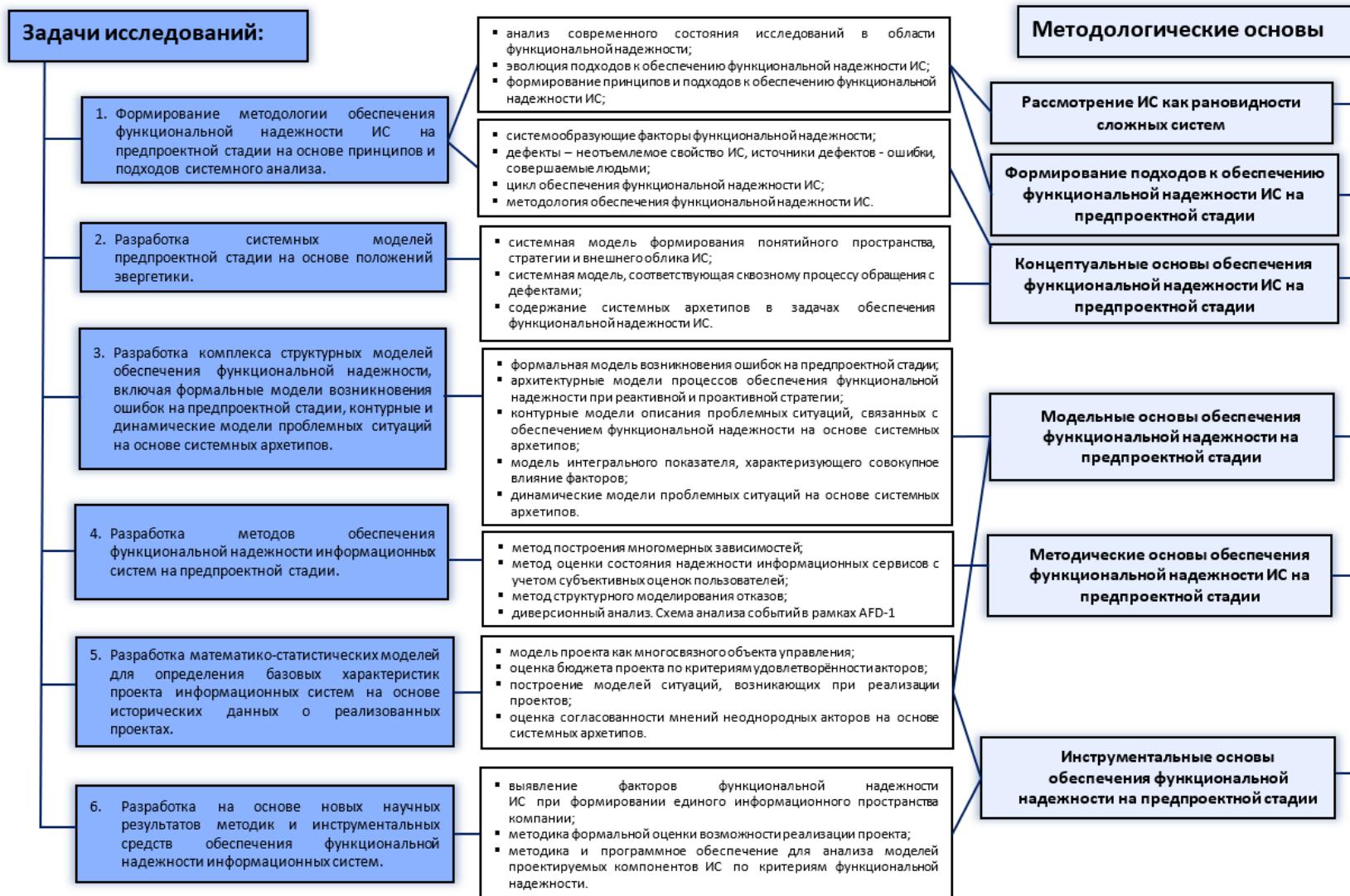


Рисунок 1.18 – Стратегия исследований, включающая совокупность задач и базовые положения исследований

## Выводы по главе 1

1) Возрастание роли ИС систем при реализации сетецентрического управления выдвигает на первый план проблему обеспечения функциональной надежности. Отмеченное обстоятельство обуславливает актуальность задачи совершенствования теоретической и методической базы обеспечения функциональной надежности ИС.

2) Эволюция подходов к обеспечению функциональной надёжности демонстрирует, что к настоящему времени теория надежности автоматических технических систем получила мощное развитие, накоплен большой опыт реализации аппаратных компонентов с высокими характеристиками качества, при этом качество программных компонентов, является недостаточным. Основным направлением повышения функциональной надежности ИС является развитие и совершенствование технологий проектирования, при этом вопросам разработки теоретических подходов к урегулированию проблемных ситуаций, неизбежно возникающих в процессе управления проектами разработки ИС, уделяется значительно меньше внимания.

3) Анализ текущего состояния исследований в области обеспечения функциональной надежности показал преобладание исследований, основанных на применении реактивного подхода. Однако данный подход не обеспечивает необходимого уровня функциональной надежности ИС. Существенно меньшее внимание уделяется исследованиям, направленным на реализацию проактивного подхода, а также на выявление и анализ дефектов, обусловленных субъективными ошибками со стороны правообладателей и разработчиков, принимающих участие в создании ИС.

4) Сформулирована цель исследования, определен круг научных и практических задач, базовые положения исследований, сформирована основная научная идея работы: комплексное исследование проблемы обеспечения функциональной надежности ИС на предпроектной стадии на основе положений эвергетики.

Основные результаты, представленные в данной главе, опубликованы в статьях автора данного исследования в разделах введение [5-7, 9, 18, 29].

## ГЛАВА 2. КОНЦЕПТУАЛЬНЫЕ ОСНОВЫ ОБЕСПЕЧЕНИЯ ФУНКЦИОНАЛЬНОЙ НАДЕЖНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ

Данная глава посвящена концептуальным основам обеспечения функциональной надежности ИС на предпроектной стадии.

Первый параграф посвящен формулированию принципов и подходов к обеспечению функциональной надежности ИС как разновидности сложных систем.

Во втором параграфе предложена методология обеспечения функциональной надежности ИС на предпроектной стадии. Рассмотрены концептуальные основы. Подробно компоненты методологии последовательно рассматриваются в соответствующих разделах диссертации.

В третьем параграфе подробно рассмотрены свойства дефектов как предмета исследования, показана эволюция дефектов на предпроектной стадии и в ходе реализации проектов разработки ИС. Разработана системная модель, соответствующая сквозному процессу обращения с дефектами на разных стадиях жизненного цикла.

В четвертом параграфе на основе положений эвергетики разработаны системные модели предпроектной стадии. Рассмотрены системообразующие факторы возникновения ошибок.

В пятом параграфе рассмотрено содержание системных архетипов в задачах обеспечения функциональной надежности ИС. Использование системных архетипов для описания проблемных ситуаций способствует их структуризации, что, в свою очередь, формирует основу для адекватного восприятия этих ситуаций.

## 2.1 Принципы и подходы к обеспечению функциональной надежности информационных систем

Обеспечение функциональной надежности ИС необходимо рассматривать с позиций управления сложными системами. Рассмотрение функциональной надежности ИС как разновидности сложных систем создает методологическую основу для научно-обоснованной адаптации подходов, принципов, методов и моделей, хорошо зарекомендовавших себя при решении задач управления сложными системами иной природы, в область обеспечения функциональной надежности ИС.

В данном разделе формируется система принципов, составляющих основу разрабатываемых концептуальных основ обеспечения функциональной надежности ИС. Данные принципы необходимы для формирования теоретических представлений, системы взглядов и выбора возможных путей информационной поддержки обеспечения функциональной надежности ИС на предпроектной стадии.

В соответствии с содержательными характеристиками и назначением, все принципы, применяемые в области обеспечения функциональной надёжности, целесообразно классифицировать на две основные группы: общесистемные принципы и принципы проектирования систем обеспечения функциональной надежности.

### *Общесистемные принципы:*

1) *Принцип системности.* Данный принцип является базовым в группе общесистемных принципов, поскольку отражает фундаментальные положения системного подхода, применимого к исследованию и проектированию систем различной природы. Система обеспечения функциональной надёжности рассматривается одновременно как сложная субъектоцентрическая система и как подсистема системы информационной поддержки реализации проекта ИС.

2) *Принцип полиморфизма.* Принцип полиморфизма акцентирует внимание на множественности форм элементов системы и разнообразии связей между ними.

Реализация данного принципа позволяет описывать одну и ту же систему с использованием различных архитектурных моделей, отражающих различные подходы к обеспечению функциональной надёжности, включая тестирование, сценарный анализ, а также концепцию «барьерного мышления».

Антагонистом принципа полиморфизма, является *принцип изоморфизма*. предполагает наличие структурной и характеристической схожести между системами различной природы. Суть этого принципа сходится к тому, что единообразие форм описания систем разной природы, схожесть структур и свойств системных характеристик обуславливает возможность применения универсальных формализованных методов анализа, основанных на использовании знаковых моделей. В контексте задач обеспечения функциональной надёжности ИС данный принцип обосновывает использование архитектурного подхода и системных архетипов как инструментов проектирования и анализа.

3) *Принцип многообразия*. Сущность данного принципа заключается в признании существования множества форм (морфизмов) ошибок, допускаемых субъектами при разработке ИС, а также вызванных этими ошибками дефектов. Частным проявлением принципа многообразия выступает многокритериальность, предполагающая необходимость учета и оптимизации различных критерииев (например, соотношения затрат на тестирование и вероятности проявления латентных дефектов). При этом один и тот же критерий (например, доля невыявленных латентных дефектов [74, 78]) может служить основанием для принятия решений, в том числе по продолжению или завершению тестирования модулей различного функционального назначения.

4) *Принцип декомпозиции*. В рамках обеспечения функциональной надёжности данный принцип означает возможность поэтапного выделения ключевых факторов, временных и пространственных характеристик возникновения дефектов, а также ресурсов и механизмов, направленных на их предотвращение или устранение. Горизонтальная декомпозиция позволяет исследовать дефекты, проявляющиеся в продуктах проектной деятельности, включая промежуточные артефакты, тогда как вертикальная декомпозиция направлена на выявление и анализ дефектов, возникающих в системе управления проектом.

5) *Принцип интеграции (композиции).* Данный принцип предполагает возможность построения совокупности моделей, способствующих выбору рациональных стратегий достижения требуемого уровня функциональной надёжности системы.

6) *Принцип эквивалентных путей достижения цели.* Учитывая неопределённость, присущую как состоянию проекта, так и характеристикам ИС на различных этапах жизненного цикла (см. модель «Конус неопределённости» [81]), невозможна реализация принципа единственности Р. Колмана при построении моделей, ориентированных на решение задач обеспечения функциональной надежности.

Предполагается наличие множества альтернативных способов достижения приемлемого уровня функциональной надёжности, реализуемых как в рамках реактивного, так и проактивного подходов. Выбор конкретного способа должен основываться на оценке результативности и эффективности в конкретных условиях реализации проекта.

7) *Принцип системной готовности.* В контексте обеспечения функциональной надёжности данный принцип заключается в том, что инцидент (латентный дефект) приводит к негативным последствиям лишь при одновременном возникновении совокупности условий, необходимых для его активации. Это, в свою очередь, может инициировать реализацию так называемого «принципа домино». Следовательно, основополагающим принципом предупреждения дефектов является исключение условий, при которых дефект может проявиться и стать триггером цепной реакции отказов.

8) *Принцип объективности.* Согласно данному принципу из множества возможных оценок следует выбирать лишь те, которые опираются на проверенные знания и достоверные эмпирические данные. Недостаток информации не может быть компенсирован допущениями, не имеющими фактической основы, поскольку подобная подмена приводит к искажению результатов и снижает достоверность выводов.

*Принципы проектирования систем обеспечения функциональной надежности:*

1) *Принцип адекватности.* Суть данного принципа заключается в том, что масштаб и характер усилий по предотвращению возникновения дефектов/парирования их проявлений должны быть адекватны тяжести последствий, связанных с проявлением дефектов. В частности, затраты на проведение испытаний, а также выбор методов и технологий тестирования, должны быть соразмерны негативным последствиям, связанным с недоступностью информационного сервиса или снижением надежности информационного продукта. При этом стратегии тестирования, реализуемые на различных стадиях жизненного цикла ИС, должны быть соотнесены со степенью сложности проверяемого объекта: по мере увеличения его сложности, должна возрастать и сложность условий проведения испытаний.

2) *Принцип согласованности.* Реализация данного принципа предполагает равнозначность задач обеспечения функциональной надёжности по отношению к другим задачам проекта. Это требует их последовательной и непрерывной реализации на всех стадиях проекта.

3) *Принцип оптимальности.* Оптимальность предполагает рациональное распределение доступных ресурсов (материальных, финансовых и интеллектуальных) между задачами, направленными на функциональной надежности ИС. Целью является достижение сбалансированного соотношения между ключевыми параметрами проекта: сроками реализации, бюджетом и потребительскими свойствами создаваемой системы.

4) *Принцип сбалансированности.* Данный принцип акцентирует внимание на необходимости координации усилий, направленных, с одной стороны, на обеспечение функциональной надёжности в рамках текущих проектных решений, а с другой – на развитие новых подходов и совершенствование существующих методов с учётом растущей сложности управляемых объектов, масштабов проектов и эволюции управленческих парадигм, применяемых к сложным системам.

5) *Принцип типизации и стандартизации.* Сущность этого принципа заключается в необходимости строгого соблюдения требований и ограничений, установленных в нормативных документах, регламентирующих реализацию проекта. В частности, это касается учёта базовых требований к потребительским характеристикам ИС [50], а также программных проектов (базовыми документами в рамках архитектурного подхода являются [44, 126]).

6) *Принцип реализуемости.* Под реализуемостью понимается необходимость выбора практико-ориентированных и ресурсно-обоснованных мер, направленных на предотвращение дефектов и снижение последствий их проявления. Эти меры должны определяться с учётом назначения ИС, с учетом возможных последствий, связанных с проявлением латентных дефектов различной природы, а также текущих ограничений на ресурсы проекта.

Ниже представлены подходы к обеспечению функциональной надежности ИС, каждый из которых базируется на принципах создания и функционирования больших и сложных систем.

1) Системный подход, выступающий в качестве методологической основы исследования, ориентирован на анализ процессов обеспечения функциональной надёжности ИС с позиций общесистемных принципов. В рамках данного подхода приоритетным становится выявление функциональной значимости отдельных подсистем в формировании свойств системы в целом, а также выявление особенностей взаимодействия системы и внешней среды.

2) Динамический подход предполагает учёт изменений в составе и содержании требований к системе под воздействием внешней среды, включая трансформацию потребностей пользователей, а также изменение объёма и доступности ресурсов проекта. В рамках данного подхода система рассматривается как динамический объект, состояние которого подвержено влиянию как внутренних, так и внешних факторов.

3) Применение структурного подхода ориентировано на выявление закономерностей в организации системы обеспечения функциональной надёжности ИС, что позволяет установить взаимосвязь между структурой и свойствами.

4) Кибернетический подход, реализуемый как методологическая основа для исследования и управления, рассматривает систему обеспечения надёжности как управляемый динамический объект. Эффективность управления в этом контексте обеспечивается на основе использования механизмов обратной связи, позволяющих корректировать поведение системы на основе информации о внутреннем состоянии объекта и состоянии внешней среды.

5) Ситуационный подход ориентирован на принятие проектных и управленческих решений в условиях неопределённости и изменчивости внешней среды. В основе ситуационного подхода анализ актуальной информации и использовании накопленного опыта в условиях изменения ресурсов проекта, состава правообладателей и т.д.

6) Ресурсно-целевой подход фокусируется на обоснованном и целенаправленном распределении ресурсов между видами деятельности, направленными на обеспечение функциональной надёжности ИС. Предполагает установление приоритетов и выработку стратегий, обеспечивающих достижение заданных целевых показателей при ограниченных ресурсах.

7) Информационный подход ориентирует на создание сложных информационных систем, реализующих комплексную обработку разноаспектной информации. В рамках этой методологии фокусом исследований являются: информационные потоки, связанные с изучением потребностей правообладателей, пользователей, их ожиданий, анализ доступных ресурсов проекта.

8) Ценностно-ориентированный подход фокусируется на создании продукта, максимально соответствующего потребностям и ожиданиям пользователей. Основная идея подхода – максимальный учет на начальных стадиях проектирования всех особенностей использования ИС с тем, чтобы в максимально возможной степени обеспечить соответствие функциональной надежности ИС требованиям, предъявляемым со стороны пользователей.

9) Реактивный подход фокусируется на анализе результатов испытаний и исторических данных, отражающих опыт эксплуатации ИС, с целью выявления ошибок, установления причин их возникновения, а также закономерностей в проявлении симптомов дефектов.

10) Проактивный подход предполагает выявление как настоящих, так и будущих проблем, связанных с обеспечением надежности, основная цель – предотвращение инцидентов. Однако, необходимо понимать, что невозможно предотвратить все инциденты до их возникновения, поэтому необходимо поддерживать разумный баланс между проактивным и реактивным подходами.

11) Барьерный подход ориентирует на разработку многослойных, эшелонированных систем защиты от агрессивных внешних воздействий. Ни один из отдельных барьеров не обеспечивает полной защиты от негативных последствий, вызванных потенциальной опасностью. Однако интеграция в единую систему барьеров позволяет повысить уровень защиты за счёт проявления системного эффекта. Концептуальной основой подхода «барьерного мышления» (*barrier thinking*) является философия «*defense-in-depth*» [200].

Рассмотрим основные критерии сложности (также рассмотрены [108, 111]) в контексте обеспечения функциональной надёжности ИС.

- *Связность*;

Сложные адаптивные системы включают множество разнотипных агентов, взаимодействие между которыми формируются связи различной силы, определяющей степень соответствия и взаимовлияния между агентами.

- *Автономность*;

Агенты функционируют без централизованного управления, обладая определённой автономностью в рамках установленных правил, принятых для системы. Высокая автономность агентов подразумевает более высокую сложность системы.

- *Эмерджентность*;

Поведение сложной системы формируется из локальных взаимодействий самоорганизующихся агентов, приводящих к возникновению глобальных структур. Эти структуры, в свою очередь, начинают влиять на поведение самих агентов посредством отрицательных (ослабляющих) или положительных (усиливающих) обратных связей.

– *Неравновесность;*

Система функционирует в условиях постоянных внешних и внутренних возмущений. Такие изменения могут приводить к колебаниям, связанным с циклическим переходом от одного состояния равновесия – к другому, при этом сила связей между элементами влияет на устойчивость системы. Понимание силы связей между элементами позволяет выявлять «узкие места» и в будущем управлять поведением сложной системы.

– *Нелинейность;*

Итоговое поведение сложной системы не является простой суммой поведения её отдельных компонентов (агентов). Незначительные возмущения могут вызвать масштабные последствия – проявление «эффекта бабочки». Возникновение автокаталитических процессов нередко обусловлено незначительными, а порой и случайными событиями.

– *Самоорганизация;*

Сложная система способна самостоятельно перестраивать структуру и поведение в ответ на изменения, восстанавливая устойчивость и предотвращая деградацию. Такая адаптивность может порождать коллективные решения и новые формы функционирования – проявления «эмурджентного разума».

– *Эволюция;*

Если рассматривать внешнюю среду системы как совокупность всех взаимодействующих с ней систем, становится очевидно, что сложные системы являются открытыми: они не только адаптируются к условиям среды, но и активно трансформируют её. Такое взаимодействие носит, как правило, необратимый характер – принятые в условиях самоорганизации решения невозможно воспроизвести повторно, поскольку исходные условия уже утрачены. Другими словами, развиваясь параллельно и асинхронно, сложные системы и их среды оказывают влияние друг на друга, т.е. коэволюционируют.

Представленные критерии, принципы и подходы к обеспечению функциональной надежности ИС являются компонентами методологии обеспечения функциональной надежности ИС на предпроектной стадии.

## 2.2 Методология обеспечения функциональной надежности информационных систем на предпроектной стадии

Концептуальная основа данного исследования опирается на следующую систему взглядов.

1) Обеспечение функциональной надежности ИС в составе систем поддержки сетецентрического управления распределенными сложными системами относится к классу задач управления в условиях динамических нечетких целей управления.

2) ИС представляют собой разновидность сложных субъектоцентрических систем. Данный факт подтверждает возможность научно-обоснованной адаптации известных подходов к модельному описанию проблемных ситуаций, возникающих при управлении сложными системами разной природы в область обеспечения функциональной надежности ИС. Модельное описание проблемных ситуаций служит основанием для информационной поддержки принятия рациональных решений по их урегулированию.

3) Критическим фактором функциональной надежности являются дефекты. Источниками дефектов являются ошибки, допускаемые субъектами, причастными к разработке и использованию ИС.

4) Основу обеспечения функциональной надежности составляет комплексное использование сведений, получаемых из разных источников: структурный анализ архитектур систем обеспечения функциональной надежности и внутреннего устройства ИС, результаты обработки исторических данных (включая метрические характеристики), связанные с проявлением дефектов разной природы, экспертные оценки субъектов, причастных к разработке ИС.

5) Функциональная надежность определяется количеством дефектов (проявившихся и латентных), возникших на разных стадиях жизненного цикла, начиная с осознания наличия проблемной ситуации, для урегулирования которой необходимо использование ИС. Моменты проявления латентных дефектов и моменты их возникновения распределены в пространстве и во времени.

Методология обеспечения функциональной надежности ИС на предпроектной стадии представлена на рисунке 2.1.

Базовые компоненты методологии обеспечения функциональной надежности ИС на ранних стадиях жизненного цикла включают в себя:

- концептуальные основы обеспечения функциональной надежности, в основе которых интеграция нормативного обеспечения [44 - 53, 152 - 154, 166-172], общенаучных подходов и принципов;
- методические основы обеспечения функциональной надежности, включающие метод построения многомерных зависимостей, метод структурного моделирования последствий отказов, метод оценки состояния функциональной надежности на основе субъективных оценок пользователей;
- модельные основы обеспечения функциональной надежности в виде комплекса моделей, предполагающего последовательную разработку системных, структурных, контурных и динамических моделей;
- инструментальные основы обеспечения функциональной надежности, включающие методики и информационные сервисы, реализованные на основе разработанных методов и моделей.

Фокусом методологии является:

- с одной стороны, выделение общих свойств ИС со сложными системами, что служит основанием для научно-обоснованной адаптации известных в теории систем подходов управления надежностью в область обеспечения функциональной надежности ИС;
- с другой стороны, выявление специфических особенностей ИС, позволяющих выделить функциональную надежность ИС в качестве самостоятельного предмета исследований, что служит основанием разработки специальных подходов, методов и инструментов исследования.

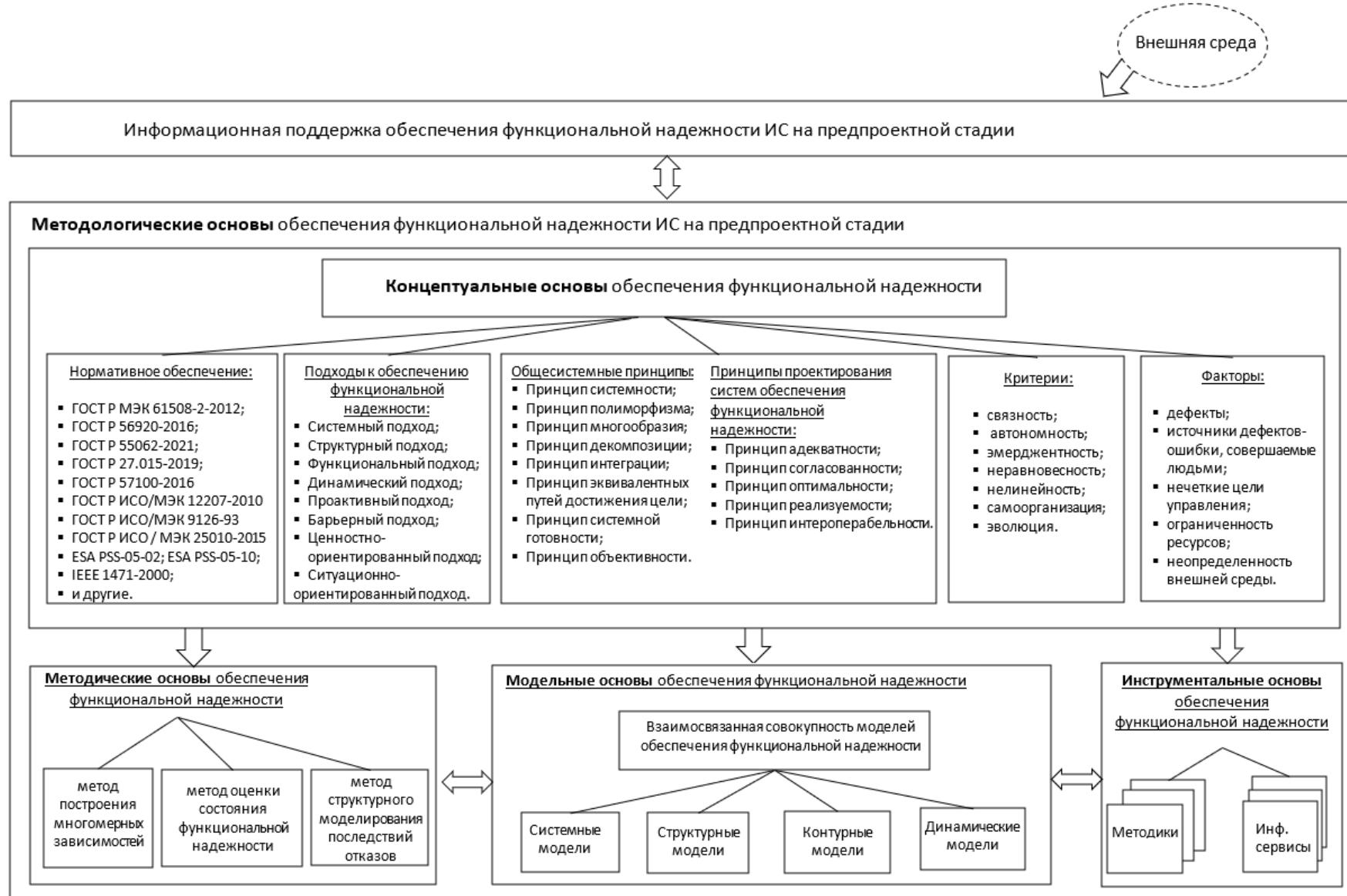


Рисунок 2.1 – Методология обеспечения функциональной надежности ИС на предпроектной стадии

На основе предложенной методологии осуществляется информационная поддержка процессов, связанных с обеспечением функциональной надёжности ИС, начиная с предпроектной стадии, что способствует своевременному выявлению и предупреждению дефектов. Подробнее наиболее существенные компоненты методологии рассматриваются в соответствующих разделах диссертации.

На рисунке 2.2 представлен цикл обеспечения функциональной надёжности, отражающий основные элементы непрерывного процесса совершенствования обеспечения функциональной надежности ИС. Данная модель построена по аналогии с Циклом Деминга<sup>7</sup> [161] в науке об управлении качеством.



Рисунок 2.2 – Цикл обеспечения функциональной надежности ИС

Содержание модели состоит в том, что по мере возрастания роли и изменения масштабов и сложности ИС, постоянно вырастают требования к функциональной надежности. Это требует постоянного развития теоретических основ, технологических и инструментальных средств обеспечения функциональной надежности.

<sup>7</sup> Цикл Деминга, или PDCA (Plan-Do-Check-Act «Планируй – Выполняй – Проверяй – Улучшай») – модель непрерывного улучшения процессов, разработанная Уильямом Эдвардсом Демингом. Данный метод помогает организациям постоянно совершенствовать свою деятельность и повышать качество продукции или услуг.

## 2.3 Принципы исследования дефектов с позиции субъектоцентрической природы информационных систем

Дефекты являются неотъемлемой составляющей субъектоцентрических систем, к числу которых относятся ИС.

Дефект - состояние ИС, которое не соответствует заявленным требованиям, спецификациям или ожиданиям пользователя (фактор, снижающий ценность информационных сервисов для потребителей).

Общей чертой дефектов, присущих ИС и объединяющих их со сложными системами иной природы, является наличие следующих характерных особенностей:

- Критическая роль субъективных ошибок и дефектов разной природы. Субъективные ошибки, а также дефекты ментального, организационного, конструктивного и технологического характера, в совокупности с особенностями архитектуры системы, являются определяющими факторами, влияющими на функциональную надёжность ИС.
- Неизбежность латентных дефектов. В субъектоцентрических системах, к числу которых относятся и ИС, наличие скрытых (латентных) дефектов различной природы неизбежно. Это положение подтверждается, в частности, в работах *J. Reason* [200, 201], где подчёркивается, что дефекты являются неотъемлемой составляющей всех субъектоцентрических систем в силу того, что человеку свойственно ошибаться.
- Зависимость между сложностью системы и характером дефектов. Характер дефектов неразрывно связан со сложностью систем: чем сложнее система, тем больше возможность возникновения в ней дефектов. Это положение обосновано, например, в документах *ESA-PSS-05-10* [150] и *ESA-PSS-05-11* [153].
- Динамическая природа дефектов. Дефекты являются динамическими объектами, форма и содержание которых зависят от стадии жизненного цикла и эволюционируют по мере перехода на иную стадию.

– Эффект наследования дефектов. Дефекты, проявляющиеся на более поздних стадиях жизненного цикла, обусловлены дефектами, возникшими на более ранних стадиях жизненного цикла. Возникновение и проявление дефектов, как правило, разделены во времени и пространстве, что значительно усложняет анализ коренных причин потери функциональности ИС. Данный феномен подробно рассматривается в [201].

– Управляемость дефектов. Дефекты могут быть объектами как реактивного (устранение последствий уже проявившихся сбоев), так и проактивного (препятствие распространения последствий) управления.

К специфическим свойствам дефектов, характерным для ИС и отличающим их от дефектов в сложных системах иной природы, относятся следующие.

– Принципиальная невозможность установления факта наличия всех дефектов. Установление наличия всех возможных дефектов в информационных системах с помощью специально организованных испытаний является теоретически и практически недостижимым. На данное ограничение указывается, в частности, в работе Г. Майерса [78], где подчёркивается невозможность исчерпывающего тестирования программного обеспечения.

– Высокая степень неопределённости. Неопределенность требований к свойствам ИС и неопределенность состояния внешней среды ИС в составе динамических распределённых вычислительно-коммуникационных систем, существенно затрудняет идентификацию дефектов на ранних этапах.

– Отсутствие/малое число измерительных данных. Недостаточность статистических и эмпирических данных, характеризующих функциональную надёжность ИС в различных режимах эксплуатации, обусловлена динамическим характером изменения состояния объектов сетецентрического управления.

Указанные особенности обуславливают необходимость применения расширенного спектра исследовательских инструментов, включая методы структурного анализа, математико-статистические методы обработки малых выборок, а также экспертно-статистические методы, позволяющие компенсировать недостаток объективных данных.

Трансформация ошибок/дефектов на предпроектной стадии (рисунок 2.3) и в ходе реализации проектов ИС означает, что при переходе на последующую фазу жизненного цикла существовавшие ранее дефекты в системе управления проектом и в продукте трансформируются в дефекты иной формы как в системе управления, так и в продукте (рисунок 2.4).

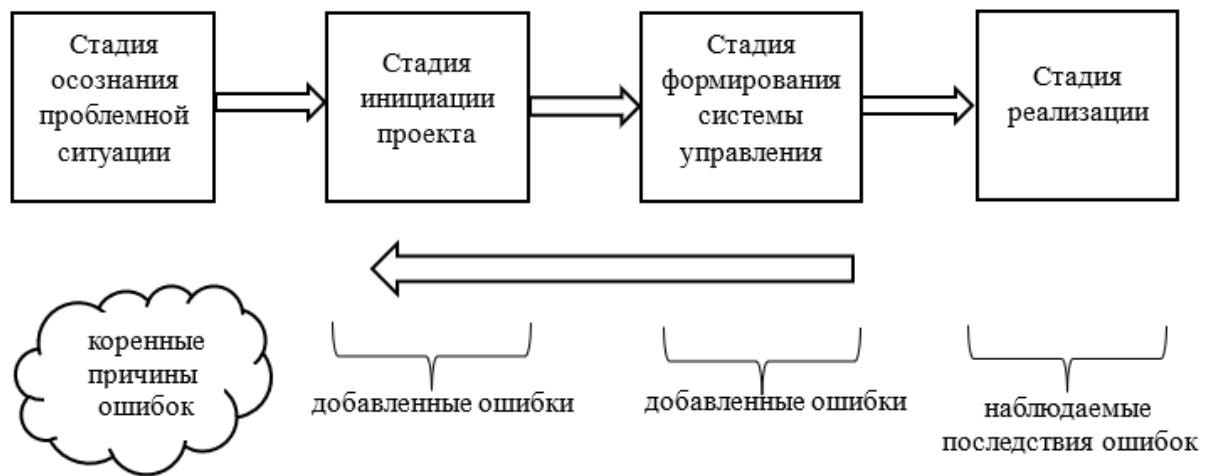


Рисунок 2.3 – Обобщенная схема трансформации ошибок/дефектов на предпроектной стадии

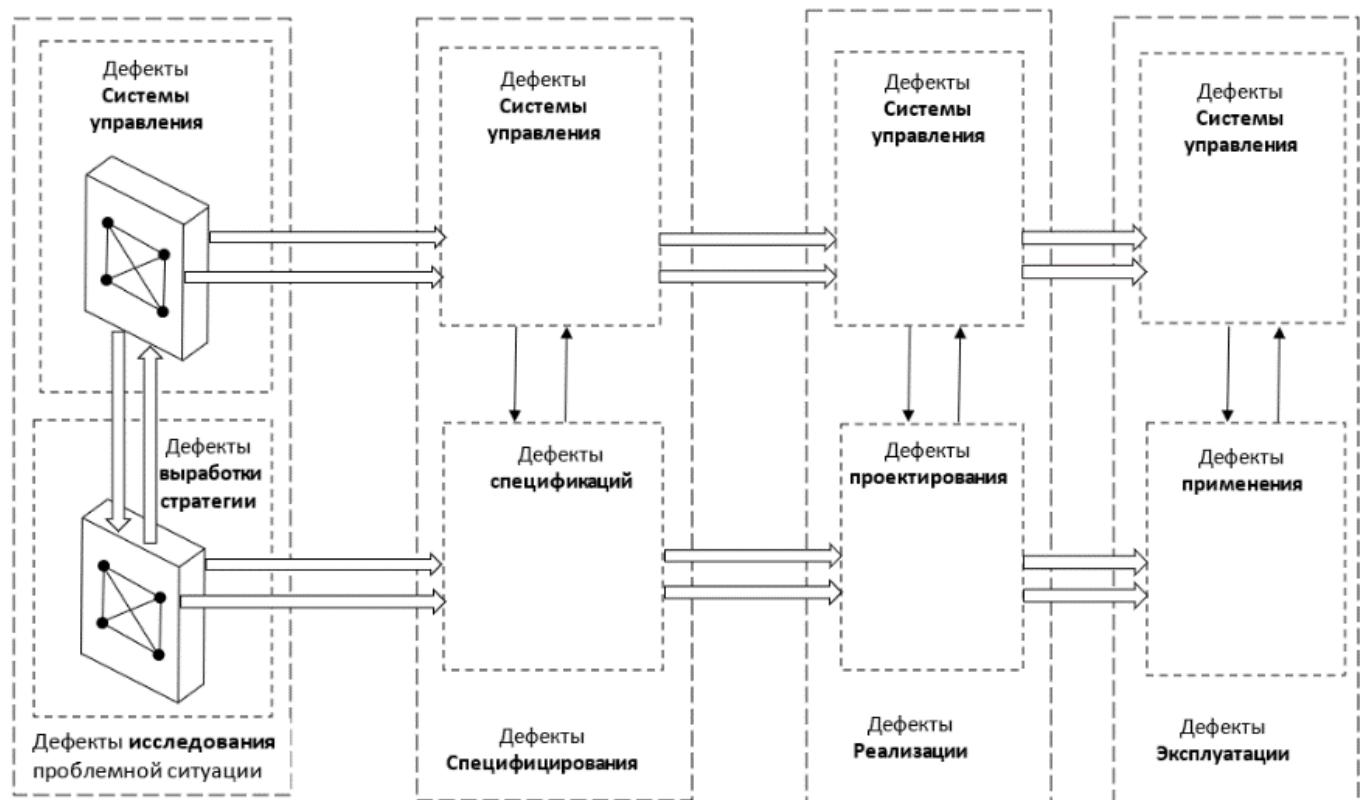


Рисунок 2.4 – Метаморфоза ошибок/дефектов

Дефекты - сложные (многосвязные) объекты управления.

Системная модель, соответствующая сквозному процессу обращения с дефектами на разных стадиях жизненного цикла, представляет собой модель взаимодействия двух компонент. Унифицированным способом представления обоих компонент на разных стадиях жизненного цикла является структурная модель, известная как «Дом качества» (*House of Quality – HoQ*) [140, 173, 187, 161].

Формальная математическая модель:

$$HoQ: \|V^i\| \times \|A^i\| \rightarrow \|W^i\| \quad (2.1)$$

где  $\|V^i\|$  – вектор первичных дефектов, соответствующий «крыльцу» *HoQ*;

$\|A^i\|$  – матрица, показывающая связь первичных и вторичных дефектов, соответствующая «комнате» *HoQ*;

$\|W^i\|$  – вектор вторичных дефектов, соответствующий «крыше» *HoQ*.

Первая компонента связана с анализом причин возникновения дефектов и видом дефектов, представлена на рисунке 2.5. Вторая компонента связана с формированием подходов к предупреждению возникновения, выявления и устранения дефектов, представлена на рисунке 2.6.

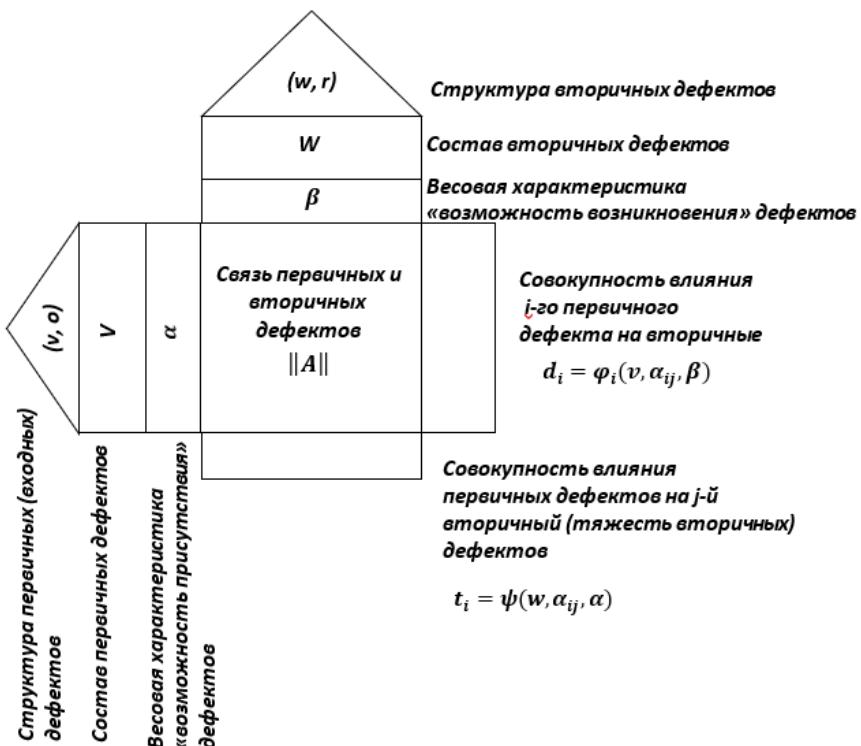


Рисунок 2.5 – Содержание *HoQ* <Дефекты>

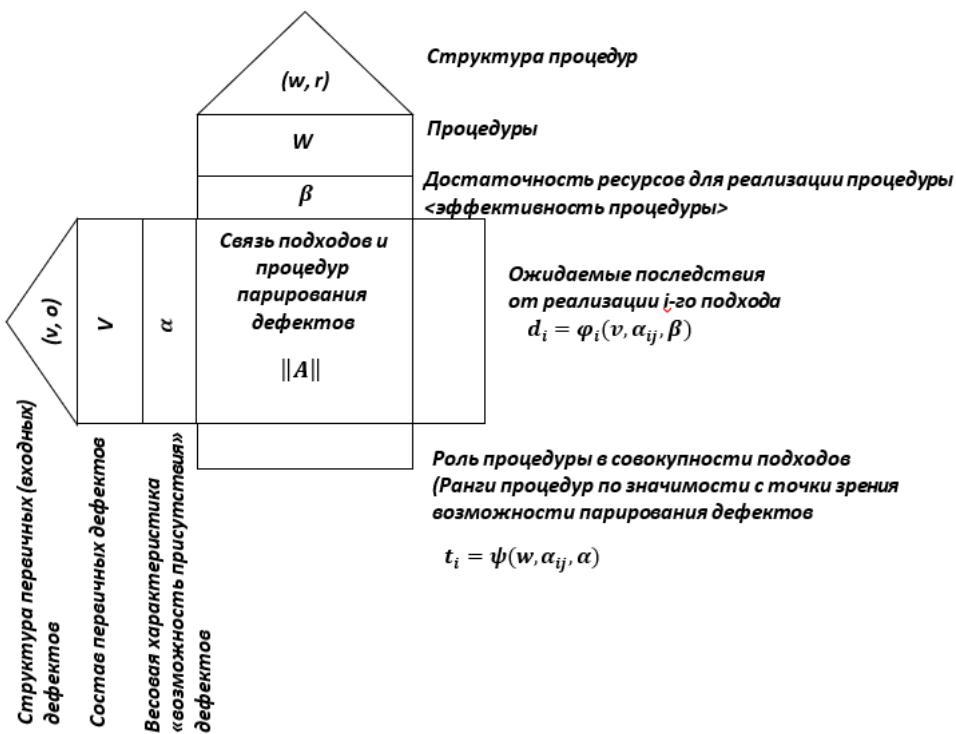


Рисунок 2.6 – Содержание *HoQ <Парирование дефектов>*

Компоненты модели характеризуют разное видение дефектов в рамках реактивного и проактивного подходов.

Системная модель, соответствующая сквозному процессу обращения с дефектами на разных стадиях жизненного цикла представлена на рисунке 2.7.

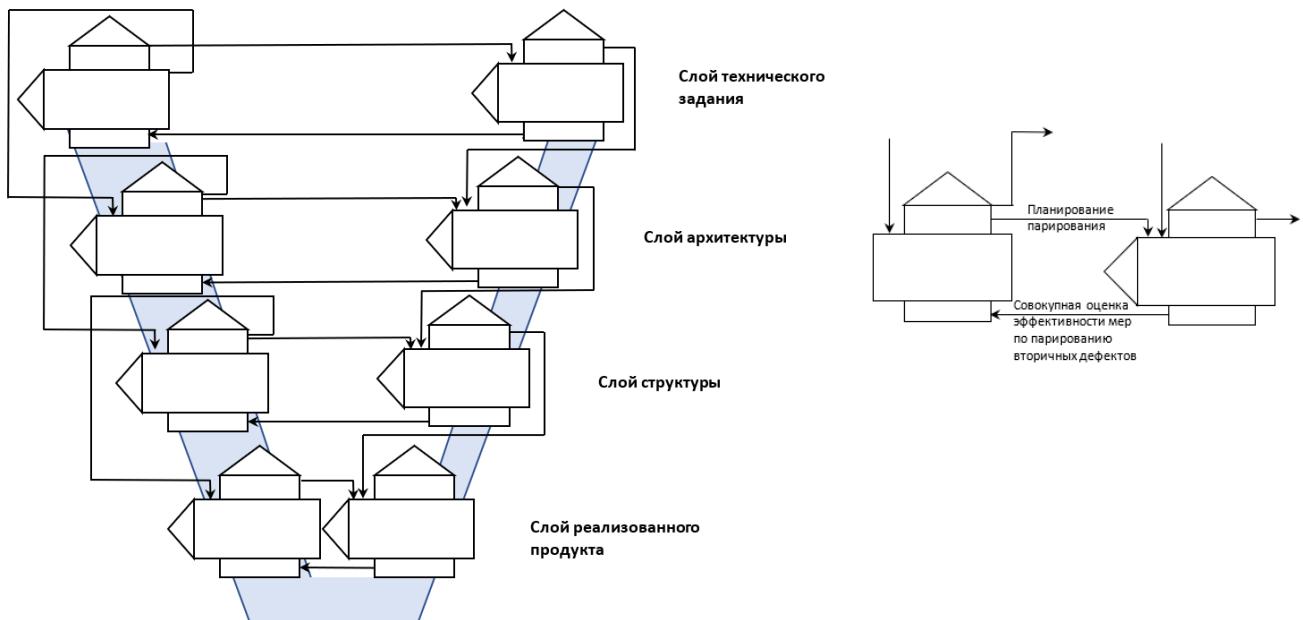


Рисунок 2.7 – Системная модель, соответствующая процессу обращения с дефектами

Таким образом, формальной моделью процесса взаимодействия двух компонент является модель, аналогичная *V*-модели жизненного цикла в сочетании с *QFD* (функция развертывания качества), реализованной в виде модели «Дом качества» - *HoQ*. Это позволяет унифицировать как описание трансформации дефектов при переходах на разные стадии жизненного цикла, так и подходы, реализующие проактивную и реактивную стратегии обращения с дефектами. Это, в свою очередь, создает основу для обеспечения сопоставимости свойств дефектов, соответствующих разному видению функциональной надежности ИС.

С одной стороны, дефекты оказывают негативное влияние на потребительские свойства ИС, с другой стороны, выявление причин возникновения дефектов способствует получению новых аксиологических знаний, создающих теоретическую основу повышения функциональной надежности ИС.

## **2.4 Системная модель формирования понятийного пространства, стратегии и внешнего облика информационной системы**

Существующие подходы к обеспечению функциональной надежности в первую очередь ориентированы на решение задач, связанных с трансформацией формальных описаний внешнего облика ИС, зафиксированных в спецификациях требований пользователей, в программные и аппаратные компоненты, обладающие требуемыми потребительскими свойствами [51, 55, 174]. Ошибки предпроектной стадии, проявляющиеся в виде дефектов в спецификациях внешнего облика ИС, как правило, не становились предметом исследований.

Инженерия требований (англ. *Requirements Engineering*) является субъектоцентрическим видом деятельности и наибольшее негативное влияние на потребительские свойства ИС оказывают ошибки, допускаемые на стадии формирования внешнего облика, обоснованием данного утверждения служит работа *J.Reason* [200].

Ошибки на стадии формирования внешнего облика (спецификации требований пользователей), обусловлены совокупностью факторов, среди которых можно выделить, например, различия в восприятии проблемной ситуации и ценностных ориентиров со стороны неоднородных акторов, вариативность и противоречивость требований к качественным характеристикам реализации как функциональных, так и нефункциональных свойств, предъявляемых различными категориями пользователей и другие.

«Проблемная ситуация – ситуация невозможности достичь какой-либо насущной цели с использованием наличных ресурсов и с учётом существующих ограничений (временных, материальных, организационных, интеллектуальных и других)».<sup>8</sup>

Основу формирования спецификации внешнего облика ИС, ориентированного на урегулирование проблемной ситуации, составляет консолидированная стратегия урегулирования проблемной ситуации, признаваемая ключевыми правообладателями. Проблемы формирования консолидированной стратегии в условиях неоднородности акторов на основе ценностно-ориентированного подхода были впервые обозначены в работах профессора В. А. Виттиха [34 - 38]. Проблемная ситуация не является объектом управления, заданным априори, а формируется в сознании людей к ней причастных и мотивированных к необходимости ее совместного урегулирования [39].

Структура процесса формирования спецификации внешнего облика ИС, представленная на рисунке 2.8, соответствует ГОСТ Р ИСО/МЭК 12207 [46].

Формирование стратегии представляет собой многоэтапный процесс, направленный на последовательное решение ряда ключевых задач, обеспечивающих целенаправленное и обоснованное управление проблемной ситуацией. К основным этапам данного процесса относятся:

- осознание наличия проблемной ситуации и её описание;
- представление проблемной ситуации в виде объекта управления;

---

<sup>8</sup> Согласно Российской энциклопедии, *bigenc.ru*

- формирование неоднородными акторами консолидированного подхода к урегулированию (т.е. целенаправленному изменению) проблемной ситуации.

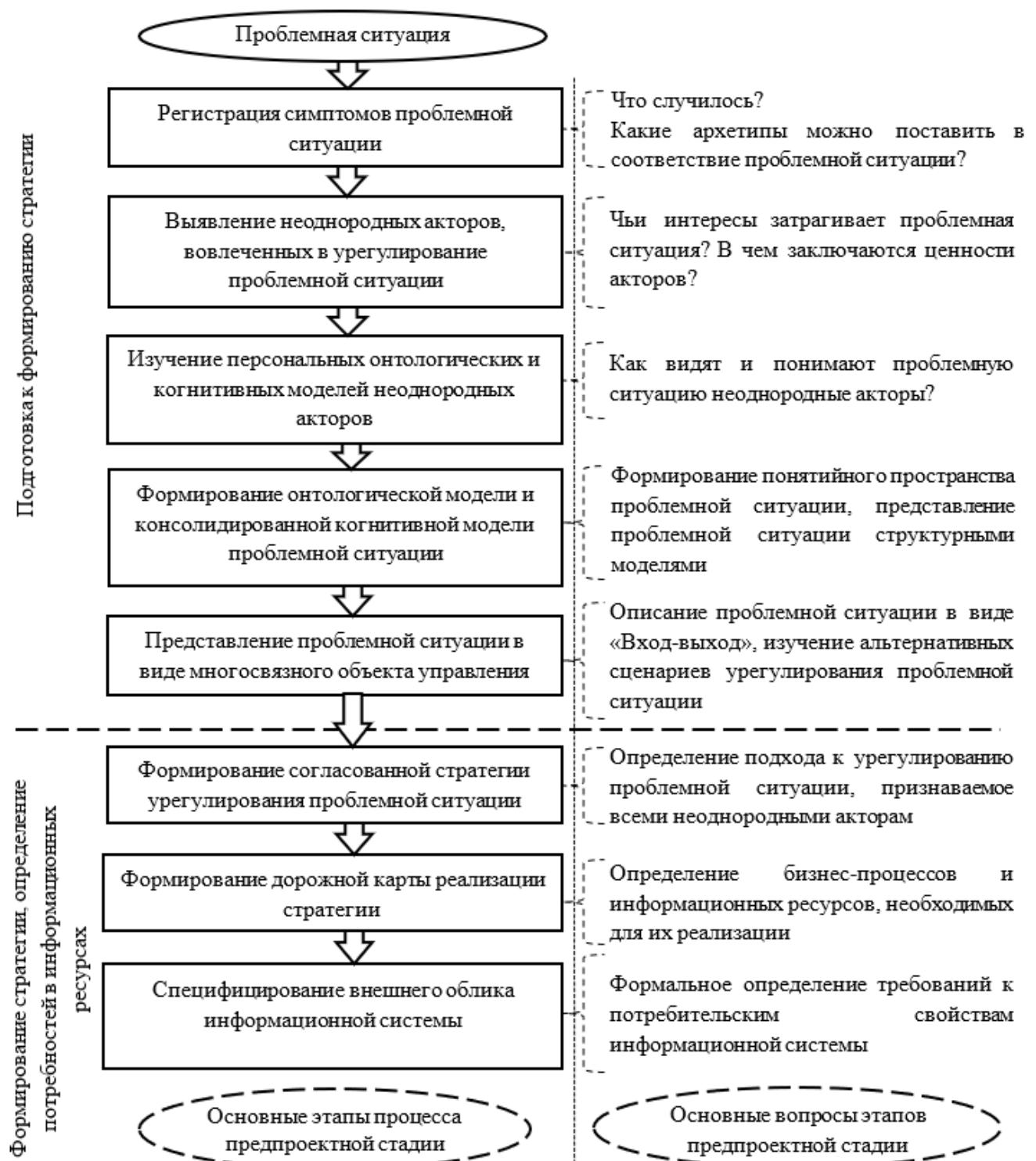


Рисунок 2.8 – Системная модель предпроектной стадии

Среди системообразующих факторов, обуславливающих возникновение ошибок на стадии выработки стратегии и впоследствии приводящих к дефектам в спецификации внешнего облика ИС, можно выделить следующие:

- Недостаток знаний о коренных причинах возникновения проблемной ситуации (часто разнесенных в пространстве и во времени относительно наблюдаемых симптомов проблемной ситуации).
- Неполнота и противоречивость понятийного отображения проблемной ситуации, обусловленная субъективизмом в восприятии проявлений проблемной ситуации неоднородными акторами
- Ограниченност персональных онтологий неоднородных акторов и обусловленная этим неполнота и различие их персональных когнитивных моделей.
- Ограниченная потенциальность моделей, используемых для описания проблемной ситуации как объекта управления.

На рисунке 2.9 представлена модель источников ошибок формирования стратегии урегулирования проблемной ситуации.



Рисунок 2.9 – Источники ошибок формирования стратегии  
урегулирования проблемной ситуации

Центральной задачей, предшествующей выработке стратегии урегулирования проблемной ситуации после осознания факта ее наличия, является понятийное и структурное представление проблемной ситуации.

Онтологическая модель проблемной ситуации строится на основе персональных онтологических моделей неоднородных акторов, вовлеченных в урегулирование проблемной ситуации.

Основой построения когнитивной (структурной) модели проблемной ситуации служат персональные когнитивные модели неоднородных акторов. В силу ограниченности знаний отдельных субъектов ни одна из персональных онтологических и когнитивных моделей не может претендовать на исчерпывающее понятийное и структурное представление проблемной ситуации (под исчерпывающим понятийным и структурным представлением понимается выделение всех, в том числе латентных, факторов возникновения проблемной ситуации, знание и понимание причинно-следственных связей между разнесенными в пространстве и во времени факторами, обуславливающими проявление проблемной ситуации).

Получение описания проблемной ситуации (понятийное и структурное представление), признаваемое всеми неоднородными акторами, вовлеченными в её урегулирование, предполагает реализацию итерационной процедуры, основанной на решении двух взаимосвязанных задач [35]:

- 1) формирование на основе персональных онтологических моделей неоднородных акторов онтологической модели ситуации;
- 2) построение на основе персональных когнитивных моделей неоднородных акторов признаваемой всеми заинтересованными сторонами когнитивной модели ситуации.

В процессе итеративного взаимодействия, основанного на дискурсе [36] между неоднородными акторами, осуществляется поэтапная коррекция их персональных онтологических и когнитивных моделей. Организация данного дискурса опирается на положения эвергетики [39, 36] и теории конвергентного управления [97], выступающей методической основой согласования различных представлений о проблемной ситуации.

Сформированная в результате когнитивная модель проблемной ситуации служит фундаментом для последующего представления проблемной ситуации в виде многосвязного объекта управления, что, в итоге, позволяет сформировать спецификации внешнего облика ИС.

На рисунке 2.10 представлена архитектура системы формирования понятийного пространства, стратегии и специфицирования внешнего облика ИС.

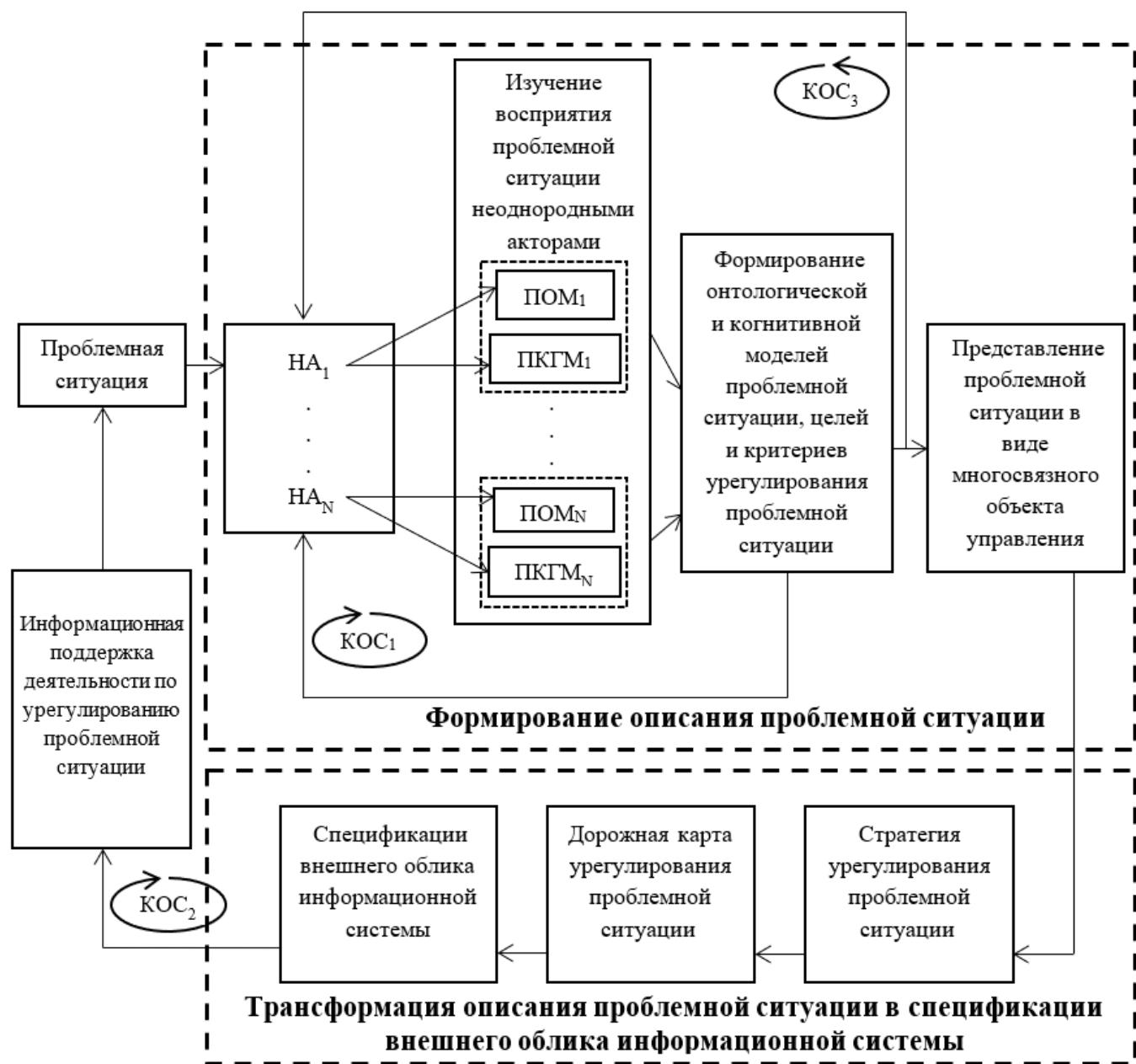


Рисунок 2.10 – Архитектура системы формирования понятийного пространства, стратегии и внешнего облика ИС

В рамках рассматриваемой архитектуры, урегулирование проблемной ситуации осуществляется на двух уровнях: в пространстве состояний проблемной ситуации и на уровне формирования нечетких целей ее урегулирования и критериев достижения целей.

Контур обратной связи КОС<sub>1</sub> соответствует дискурсу, результатом которого является построение признаваемой всеми неоднородными акторами, вовлеченными в урегулирование проблемной ситуации, когнитивной модели.

Контур обратной связи КОС<sub>2</sub> соответствует деятельности, направленной на урегулирование проблемной ситуации в пространстве ее состояний.

Контур обратной связи КОС<sub>3</sub> соответствует корректировке персональных онтологических и когнитивных моделей неоднородных акторов проблемной ситуации, целей и критериев ее урегулирования с учетом результатов имевшей место деятельности по урегулированию проблемной ситуации.

Таким образом, к числу причин возникновения дефектов в спецификации внешнего облика помимо ошибок, обусловленных мыслительной деятельностью субъектов, можно отнести недостаток знаний об истинных причинах возникновения проблемной ситуации, ограниченность понятийного отображения проблемной ситуации, недостаточная потенциальность моделей, используемых для представления проблемной ситуации в виде многосвязного объекта управления.

Представление проблемных ситуаций в виде последовательности ключевых вопросов «Что случилось?», «Чьи интересы затрагивает случилось?», «Как видят и понимают проблемную ситуацию неоднородные акторы?» служит основанием для организации целенаправленного обсуждения проблемной ситуации заинтересованными сторонами. Кроме того, в ходе дискурса формируется понятийное пространство проблемной ситуации, позволяющее сформировать единое понимание ее содержания. Это, в свою очередь, создает основу для формирования обоснованных требований к ИС, признаваемых как заказчиком, так и исполнителем, а также интегральных показателей с учетом влияния факторов, как способствующих, так и препятствующих повышению функциональной надежности ИС.

Получение ответов на последовательность вопросов предпроектной стадии создает структурные рамки проекта, которые определяют структуру, организацию и взаимосвязи понятий в рамках ИС. Структурные рамки задают границы, в которых могут находиться функциональные и нефункциональные требования к ИС. Это, в свою очередь предопределяет цели, сроки, бюджет и ресурсы проекта.

## **2.5 Содержание системных архетипов в задачах обеспечения функциональной надежности информационных систем**

Анализ подходов к обеспечению функциональной надежности ИС показал, что в настоящее время большинство исследований сосредоточено на разработке и совершенствовании методов проектирования, в то же время теоретические аспекты разрешения проблемных ситуаций, которые неизбежно возникают при реализации проектов разработки ИС, остаются менее изученными.

В научной литературе на концептуальном уровне представлены типовые проблемные ситуации, возникающие при управлении сложными системами различной природы, обозначаемые термином «системные архетипы» [138].

Термин «архетип» (от греч. *arche* – начало и *typos* – образ) трактуется как «начальный образ», формальный шаблон, отражающий устойчивую структуру проблемы, идея.

В ряде исследований [84, 92, 97, 98, 107] подчёркивается значимость архетипов как инструмента, позволяющего структурировать типовые по содержанию управленческие проблемы, характерные для функционирования сложных систем. Отмечается, в [97] «...Архетипы в управлении – это грань между хаосом и порядком. Построение архетипа - предвестник успеха любого дела...».

Особого внимания заслуживает взаимосвязь между архетипным представлением проблемных ситуаций и аксиологическим знанием субъектов, вовлечённых в реализацию проектов ИС. По мере накопления опыта и развития профессиональных компетенций у разработчиков формируется способность к

архетипному описанию проблемных ситуаций, возникающих в процессе реализации проектов ИС. Это отражает специфику управления сложной субъектоцентрической системой, функционирующей в условиях неопределенности, где традиционные подходы, ориентированные на управление в условиях четких целей и ограничений на решения, оказываются малоприменимыми.

Использование системных архетипов для описания проблемных ситуаций способствует их логической структуризации, создавая тем самым условия для адекватного восприятия проблемной ситуации и последующего формирования рациональных стратегий по урегулированию.

Результаты анализа структурированных проблемных ситуаций формируют основу для построения формализованных знаковых моделей (от когнитивных до математических).

Наличие таких моделей позволяет проводить оценку последствий альтернативных организационных решений и, соответственно, снижать вероятность проявления латентных дефектов организационной природы, способных привести к наиболее тяжелым негативным последствиям [201].

В дальнейшем рассмотрим содержание известных системных архетипов применительно к задачам обеспечения функциональной надежности ИС.

*Архетип 1 «Задержка действий».* Графическая модель, отражающая содержание данного архетипа, представлена на рисунке 2.11.



Рисунок 2.11 – Графическая модель, соответствующая архетипу «Задержка действий»

Применительно к задачам обеспечения функциональной надёжности ИС примером данного архетипа могут служить типовые ситуации, возникающие на стадиях внедрения и эксплуатации, когда потребительские свойства ИС не соответствуют актуальным потребностям системы управления. Такие несоответствия отражают рассогласование между ожидаемыми и фактическими характеристиками системы.

В качестве примеров можно привести наличие невостребованных функциональных возможностей, неудовлетворительные характеристики надежности ИС и другие аналогичные проявления.

Основными причинами подобных отклонений, как правило, выступают субъективные ошибки различной природы, приводящие к возникновению организационных, проектно-конструкторских и технологических дефектов. Так, одним из примеров организационного дефекта является игнорирование различий в потребностях и желаниях различных категорий пользователей при проведении предпроектного обследования. Процесс выявления причин возникновения дефектов и разработки соответствующих корректирующих воздействий требует временных затрат, что обуславливает наличие задержки между моментом обнаружения несоответствия и реализацией изменений. После внедрения корректирующих мер осуществляется повторное сопоставление текущих потребительских характеристик ИС с изменившимися информационными потребностями системы управления.

Однако, возможна ситуация, что в течение времени, затраченного на корректировку, требования со стороны системы управления претерпят изменения – как по объективным, так и по субъективным причинам. В результате запускается повторный цикл адаптации, что формирует колебательный процесс, снижающий устойчивость и предсказуемость функционирования системы в целом.

*Архетип 2 «Пределы роста».* Графическая модель, отражающая содержание данного архетипа, представлена на рисунке 2.12.

В контексте задач обеспечения функциональной надёжности ИС данный архетип может быть проиллюстрирован следующим примером.

С одной стороны, осуществляется стремление к развитию и совершенствованию потребительских свойств ИС в соответствии с изменяющимися характеристиками как управляемого объекта, так и внешней среды (развивающий цикл, действия ведущие к улучшению). С другой стороны, реализация новых функциональных компонентов ИС сопровождается увеличением масштабов и сложности, что, в свою очередь, усиливает неопределенность состояния систем информационной поддержки.



Рисунок 2.12 – Графическая модель, соответствующая архетипу «Пределы роста»

Это приводит к возникновению ошибок и латентных дефектов. Наличие дефектов негативно сказывается на потребительских свойствах ИС, образующих единое информационное пространство, тем самым ограничивая возможности дальнейшего улучшения ИС (стабилизирующий цикл, условия, препятствующие улучшению).

**Архетип 3 «Подмена проблемы».** Графическая модель, отражающая содержание данного архетипа, представлена на рисунке 2.13.

Применительно к проблематике функциональной надежности ИС содержание этого архетипа состоит в том, что использование краткосрочных (симптоматических) решений, быстро дающих результат, приводит к сокращению поиска фундаментальных решений. Ограниченность подхода к обеспечению функциональной надёжности, основанного преимущественно на регистрации и анализе симптомов нежелательных событий, заключается в его реактивной природе.

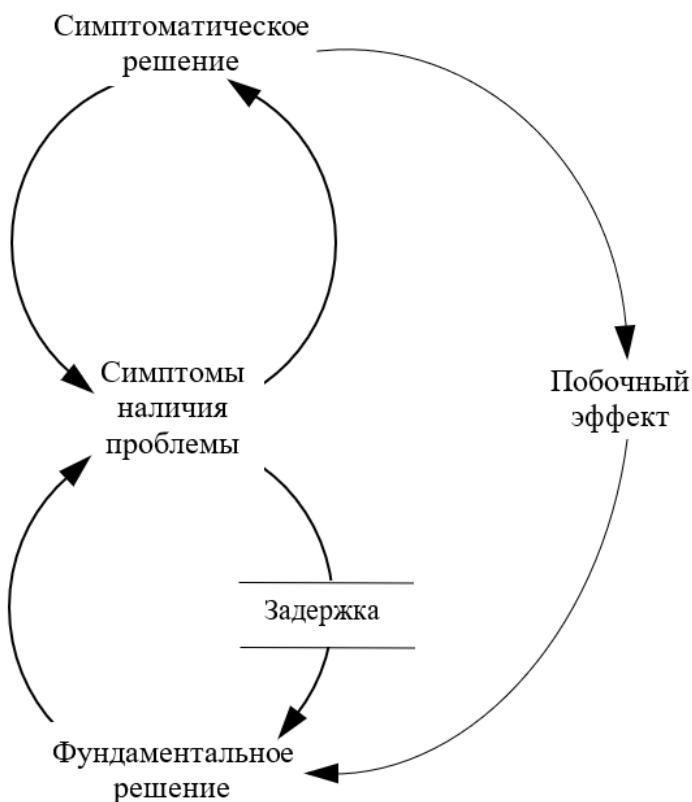


Рисунок 2.13 – Графическая модель, соответствующая архетипу «Подмена проблемы»

Такой подход не позволяет, посредством фундаментальных решений, предотвратить возникновение коренных дефектов иной природы, за исключением тех, которые были выявлены с использованием методов анализа первопричин (RCA). При этом именно такие коренные дефекты нередко становятся источником вторичных дефектов, которые продолжают латентно существовать в системе и не проявляются до наступления определённых условий в процессе эксплуатации (побочный негативный эффект подхода).

*Архетип 4 «Размытие целей».* Графическая модель, отражающая содержание данного архетипа, представлена на рисунке 2.14.

Примером архетипа в рамках формирования единого информационного пространства сетецентрического управления является неопределенность среды использования ИС, а также семантики информационных запросов. Неопределенность требований обуславливает, например, неопределенность подходов к организации испытаний (тестированию) ИС.

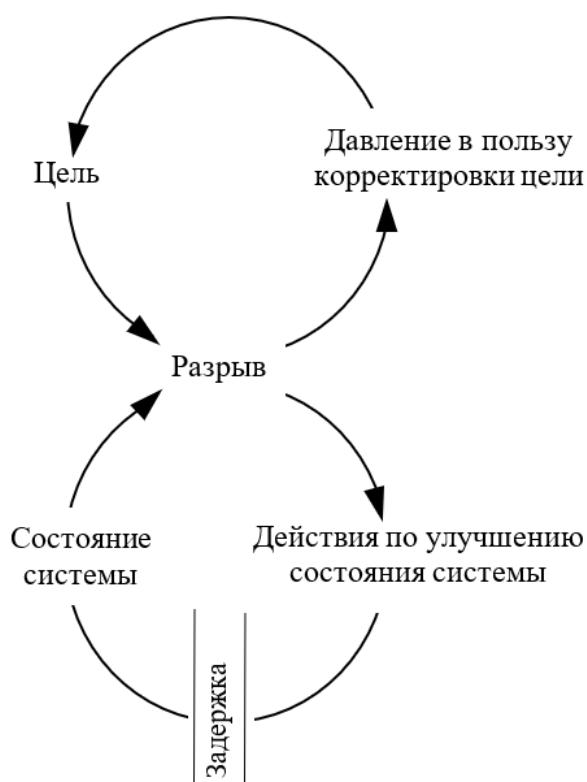


Рисунок 2.14 – Графическая модель, соответствующая архетипу «Размывание целей»

В рамках рассмотрения локальных ИС примером данного архетипа в контексте обеспечения функциональной надёжности может служить ситуация, при которой различные подразделения (целевые группы) формируют несовпадающие требования к алгоритмам обработки данных. Одни пользователи настаивают на высокой скорости обработки с минимальным набором функций, другие – на расширенной аналитике, даже в ущерб производительности. В условиях такой противоречивости (разрыв) разработчик, ориентируясь на собственные субъективные представления о приоритетности требований, фиксирует в техническом задании (цель) решение, ориентированное на производительность, рассчитывая, что функциональные ограничения будут восприняты как допустимые после ввода системы в эксплуатацию (действия по улучшению состояния системы).

После завершения работ и временной задержки (задержка) заказчику передаётся система, которая, с точки зрения разработчика, является оптимально сбалансированной (состояние системы). Однако пользователи, заинтересованные в

аналитических возможностях, оценивают систему как неудовлетворительную. В результате, в соответствии с договорными обязательствами (давление в пользу корректировки цели), разработчик вынужден срочно доработать программный продукт и устраниТЬ замечания. Возникающие в процессе доработки дефекты и ограниченность сроков приводят к снижению функциональной надёжности, что в конечном итоге приводит к неудовлетворенности и к сокращению ресурсов, выделяемых на проект.

*Архетип 5 «Эскалация (расширение)».* Графическая модель, отражающая содержание данного архетипа, представлена на рисунке 2.15.



Рисунок 2.15 – Графическая модель, соответствующая архетипу «Эскалация»

Стремление ускорить урегулирование проблемной ситуации приводит к расходованию ресурсов на деятельность, дающую быстрый результат за счет устранения дефектов, являющихся непосредственными причинами негативных событий. Это сокращает ресурсы, которые могут быть выделены на выявление коренных причин дефектов и создание барьеров, препятствующих их проявлению. Выявление коренных причин дефектов более значимо, нежели устранение непосредственных дефектов, однако эффект от устранения коренных причин проявляется позднее. В результате имеет место конкуренция за ресурсы со стороны системы оперативной ликвидации непосредственных дефектов и системы выявления и устранения коренных причин дефектов.

Примером данного архетипа в контексте обеспечения функциональной надёжности может служить ситуация, возникающая при внедрении информационных технологий в организационную среду. С одной стороны, это

способствует усилению позиций отдельных сотрудников за счёт повышения эффективности выполнения ими бизнес-задач (*контура Деятельность A – Результат A*). С другой стороны, такое технологическое обновление вызывает перераспределение должностных полномочий, в результате чего у другой группы сотрудников сокращаются как формальные, так и неформальные зоны влияния (*контура Деятельность B – Результат B*). В ответ на снижение значимости своей роли представители группы B инициируют приобретение и внедрение новых, более продвинутых информационных технологий, функциональность которых существенно превышает реальные информационные потребности текущих бизнес-процессов. Подобная динамика запускает «IT-гонку», итогом которой становится неэффективное расходование ресурсов и снижение функциональной надёжности из-за усложнения ИТ-инфраструктуры без обоснованной необходимости.

**Архетип 6 «Успех к успеху».** Графическая модель, отражающая содержание данного архетипа, представлена на рисунке 2.16.

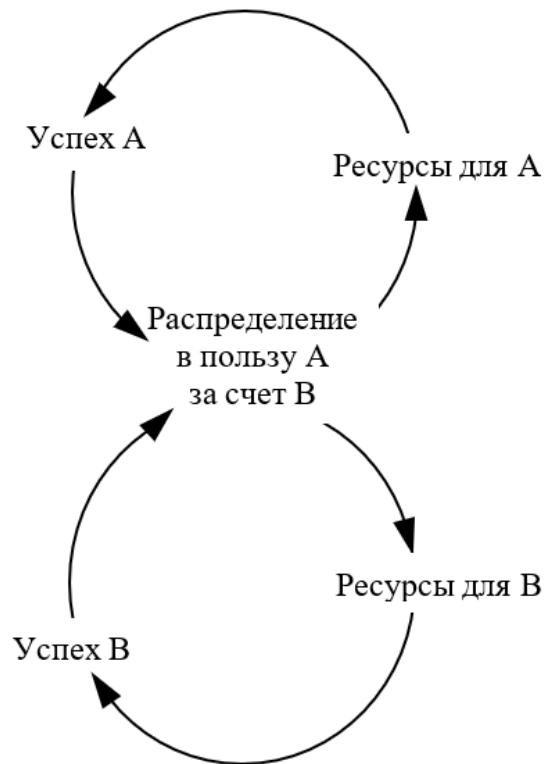


Рисунок 2.16 – Графическая модель, соответствующая архетипу «Успех к успеху»

В контексте обеспечения функциональной надёжности характерным проявлением данного архетипа является ситуация, при которой улучшение качества спецификаций требований – пользовательских, системных, функциональных и других способствует снижению количества дефектов в конечных продуктах. Это, в свою очередь, приводит к сокращению расходов на устранение ошибок (*контур Ресурсы для В – Успех В*), благодаря чему появляется возможность направить высвободившиеся ресурсы (*механизм распределения в пользу А за счёт В*) на развитие методической базы, совершенствование инструментальных средств и повышение квалификации специалистов в области обеспечения качества ИС (*контур Ресурсы для А – Успех А*).

**Архетип 7 «Трагедия общих ресурсов».** Графическая модель, отражающая содержание данного архетипа, представлена на рисунке 2.17.



Рисунок 2.17 – Графическая модель, соответствующая архетипу «Трагедия общих ресурсов»

Содержание этого архетипа сводится к тому, что недостаточность общих, выделяемых ресурсов на практическую и научную деятельность не позволяет получить значимых результатов по урегулированию проблемной ситуации.

Результатом этого является разочарование акторов получаемыми результатами и, как следствие, прекращение всякой деятельности по урегулированию проблемной ситуации.

Примером реализации данного архетипа в области обеспечения функциональной надёжности может выступать следующая ситуация. В условиях ограниченного бюджета и длительности реализации проекта руководство организации – Исполнителя принимает решение о перераспределении ресурсов в пользу разработки ИС (*контур Индивидуальная деятельность B*), при этом существенно снижая объёмы, выделяемые на проведение комплексного тестирования (*контур Индивидуальная деятельность A*). Основанием для такого решения является известный тезис о том, что тестирование является затратным мероприятием и не повышает качества программного продукта [78]. Ограничность ресурсов на испытания приводит к передаче заказчику продукта с многочисленными латентными дефектами (*Совокупная деятельность*). Повторяющиеся случаи низкого качества поставляемых продуктов спустя определённый период времени (*Задержка*) вызывают снижение спроса, что проявляется в уменьшении количества и объёма заказов (*Предельная ёмкость ресурса*) и, как следствие, снижении уровня доходов сотрудников (*Выгода от индивидуальной деятельности*).

Если не будет предпринято системное вмешательство для разрыва данного замкнутого цикла, это может привести к ликвидации организаций – Исполнителя и утрате профессиональной репутации её специалистов.

*Архетип 8 «Симптоматические решения».* Графическая модель, отражающая содержание данного архетипа, представлена на рисунке 2.18.

Содержание данного архетипа сводится к тому, что субъекты в условиях ограниченности ресурсов используют паллиативные подходы, лишь только скрывающие симптомы проблемной ситуации. Результатом «симптоматических решений» является нерациональное, с точки зрения урегулирования проблемной ситуации расходование ресурсов.

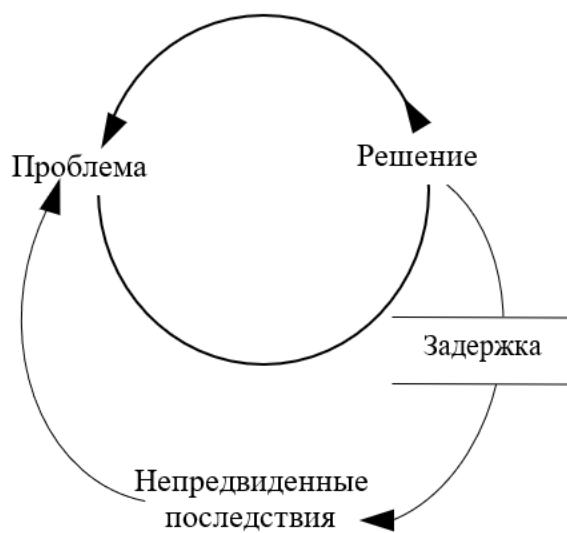


Рисунок 2.18 – Графическая модель, соответствующая архетипу «Симптоматические решения»

В качестве примера реализации данного архетипа в контексте обеспечения функциональной надёжности может служить ситуация, при которой, в условиях жёстких временных ограничений (проблема), принимается решение о преждевременном переходе к этапу кодирования программного продукта (решение), минуя полноценное предпроектное обследование. Подобный подход приводит к множественным доработкам и переработкам, возникновению межличностных напряжений в команде разработчиков (непредвиденные последствия), а также к превышению запланированного бюджета и сроков реализации проекта.

В итоге это оборачивается срывом проекта (задержка), когда заказчик не получает в установленные сроки необходимые информационные ресурсы для обеспечения своих бизнес-процессов.

**Архетип 9 «Рост и недоинвестирование».** Графическая модель, отражающая содержание данного архетипа, представлена на рисунке 2.19.

Содержание данного архетипа сводится к тому, что урегулирование проблемной ситуации можно ускорить, если своевременно и в нужных объемах вкладывать средства в развитие как технологической, так и научной базы, а также обучение субъектов, задействованных в урегулировании проблемной ситуации.



Рисунок 2.19 – Графическая модель, соответствующая архетипу «Рост и недоинвестирование»

Одним из примеров реализации данного архетипа является следующая ситуация: в условиях цифровой экономики и стремительного увеличения интереса к технологиям «Умного производства» наблюдается рост потребности в формировании цифровой экосреды на уровне предприятий.

Это, в частности, выражается в необходимости интеграции локальных информационных систем (контур «Рост производства – спрос»). Заказчик требует высокого уровня надежности цифровой экосреды и устанавливает жесткие ограничения на сроки проектов. На основе этих критериев оцениваются инвестиционные потребности.

**Архетип 10 «Случайные противники».** Графическая модель, отражающая содержание данного архетипа, представлена на рисунке 2.20.

Сущность данного архетипа заключается в следующем. Изначально два объекта вступают во взаимодействие, исходя из согласованных, взаимовыгодных целей, предполагая, что синергетический эффект от объединения усилий позволит

достичь результатов, недостижимых в условиях автономного функционирования каждого из них. Однако в процессе взаимодействия возникает ситуация, при которой одна из сторон (первый объект) совершает действие, не имеющее умысла нанести ущерб, но воспринимаемое другой стороной (вторым объектом) как отклонение от ранее установленных договорённостей и как поведение, приносящее одностороннюю выгоду первому объекту за счёт интересов второго.

При этом первый объект, не осознавая потенциально негативных последствий своих действий с точки зрения второго участника взаимодействия, не предпринимает шагов для согласования позиций. Второй объект, не вступая в конструктивный диалог, реагирует на данную ситуацию посредством действий, направленных на нанесение ущерба первому объекту. Это, в свою очередь, провоцирует ответную реакцию — первый объект предпринимает симметричные или усиливающие меры, направленные против второго объекта. В результате формируется нарастающий конфликтный процесс, который можно охарактеризовать как эффект эскалации.

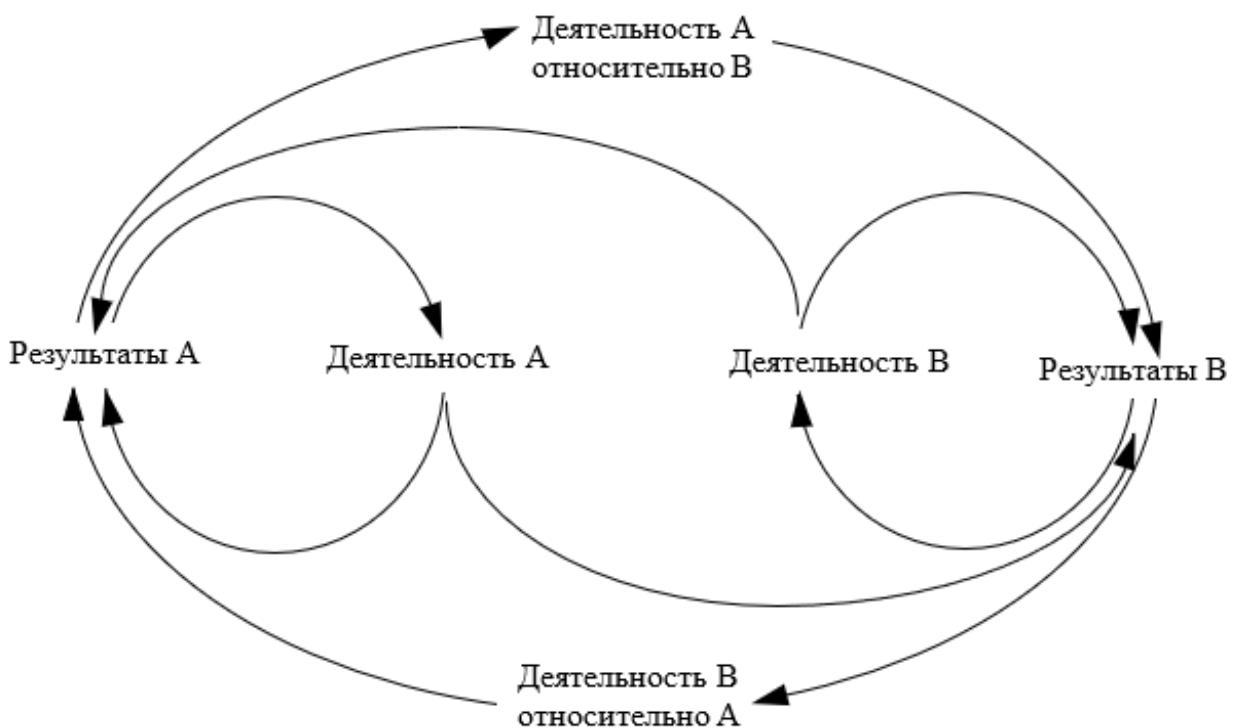


Рисунок 2.20 – Графическая модель, соответствующая архетипу «Случайные противники»

Таким образом, использование системных архетипов для описания проблемных ситуаций способствует их логической структуризации, что, в свою очередь, формирует основу для адекватного восприятия и последующей выработки обоснованных и рациональных подходов к их урегулированию. С прикладной точки зрения ограничением применения системных архетипов является их преимущественно качественный характер, в них не представлена структура системы управления и параметрические зависимости прямых и перекрестных связей, имеющих место в системе управления.

## **Выводы по главе 2**

1) Сформулированы принципы и подходы к обеспечению функциональной надежности ИС как разновидности сложных систем. Рассмотрение функциональной надежности ИС как разновидности сложных систем создает методологическую основу для научно-обоснованной адаптации подходов, принципов, методов и моделей, хорошо зарекомендовавших себя при решении задач управления сложными системами иной природы, в область обеспечения функциональной надежности ИС.

2) Предложена методология обеспечения функциональной надежности ИС на предпроектной стадии, включающая в себя: концептуальные основы обеспечения функциональной надежности, методические основы обеспечения функциональной надежности, модельные основы обеспечения функциональной надежности, а также инструментальные основы обеспечения функциональной надежности, включающие методики и информационные сервисы. В рамках системного подхода разработаны концептуальные основы обеспечения функциональной надежности, представляющие собой интеграцию нормативного обеспечения, общенаучных подходов и принципов.

3) Разработана системная модель, соответствующая сквозному процессу обращения с дефектами на разных стадиях жизненного цикла, которая объединяет в себе *V*-модель жизненного цикла и подход *QFD* (функции развертывания качества), реализованный в виде модели «Дом качества» - *HoQ*. Модель обеспечивает единый формат описания преобразований дефектов при переходах на разные стадии жизненного цикла.

4) Предложена совокупность системных моделей процессов предпроектной стадии разработки ИС. Показано, что дефекты в спецификациях внешнего облика обусловлены как особенностями когнитивных моделей субъектов, причастных к формированию требований, так и организацией формирования требований, в совокупности выделенные причины формируют системную составляющую дефектов в спецификации внешнего облика ИС.

5) Рассмотрено содержание системных архетипов в задачах обеспечения функциональной надежности ИС. Описание проблемных ситуаций посредством системных архетипов способствуют её структуризации, создает предпосылки для её адекватного восприятия и как следствие выработка рациональных подходов к её урегулированию. Полученные при исследовании структурированных проблемных ситуаций результаты создают основу для построения в различной степени формализованных знаковых моделей. Поскольку дефекты начинают возникать на предпроектной стадии, важно распознать ситуационную проблему для чего и предложен аппарат системных архетипов.

Основные результаты, представленные во второй главе, опубликованы в статьях автора данного исследования в рецензируемых печатных изданиях из перечня, утвержденного ВАК [22, 16, 20] и в материалах международных конференций [18].

## ГЛАВА 3. МОДЕЛИ ФАКТОРОВ, ПРИВОДЯЩИХ К ВОЗНИКНОВЕНИЮ ДЕФЕКТОВ И ОШИБОК НА ПРЕДПРОЕКТНОЙ СТАДИИ

В третьей главе рассматриваются вопросы построения моделей факторов, определяющих функциональную надежность ИС на предпроектной стадии.

Первый параграф настоящей главы посвящен разработке взаимосвязанной совокупности структурных моделей процессов обеспечения функциональной надежности ИС. Наличие архетипа позволяет построить разные модели для разных стратегий борьбы с дефектами. Полученные результаты создают основу для исследования свойств архитектур известными формальными методами.

Во втором параграфе представлена формальная модель ошибок предпроектной стадии, выполнена ее верификация.

В третьем параграфе представлена модель формирования структуры интегрального показателя, характеризующего совокупное влияние факторов функциональной надежности ИС.

В четвертом параграфе представлены контурные и динамические модели проблемных ситуаций основу которых составили системные архетипы. Выполнен анализ типовых ситуаций, связанных с обеспечением функциональной надежности ИС на основе архетипа «Пределы роста».

### 3.1 Структурные модели процессов обеспечения функциональной надежности информационных систем

В современных исследованиях большое внимание уделяется вопросам обеспечения функциональной надежности ИС [157, 164, 165, 185]. Критическим фактором функциональной надежности ИС является организация системы предупреждения и устранения дефектов разной природы.

В настоящем разделе представлена модель, позволяющая с единых позиций описывать разные стратегии предотвращения и устранения дефектов в ИС из-за дефектов в организации проекта.

Концептуальную основу модели составляет следующее: качество программного продукта определяется качеством организации и управления реализацией проекта. Данное утверждение соответствует рамочным рекомендациям стандарта ESA PSS-05-11 [153].

Методическую основу составляет адаптация метода, предназначенного для моделирования на качественном уровне динамики состояния системы [194] в область предотвращения возникновения и устранения дефектов в ИС при разных стратегиях управления проектом.

Фокусом метода является соотношение

$$\overrightarrow{c_{k+1}} = \vec{c}_k * \|H\| \quad (3.1)$$

где  $\overrightarrow{c_{k+1}}$  – вектор состояния узлов знаково-ориентированного графа на (k+1)-й итерации ( $k=0,1,2, \dots$ )

$\|H\|$  – матрица, характеризующая отношения между узлами знаково-ориентированного графа.

Постулируются следующие положения:

- Компонентами вектора состояний могут быть лишь элементы множества  $\{-1, 0, 1\}$ . Значение «1» соответствует положительному изменению состояния узла графа (концепта); «0» соответствует неизменному состоянию; «-1» соответствует ухудшению состояния.

- Если в ходе расчетов значение компоненты вектора состояний превышает значение «1», оно заменяется на значение «1».
- Если в ходе расчетов значение компоненты вектора состояний принимает значение меньше, чем «-1», оно заменяется на значение «-1».
- Если в ходе расчетов значение компоненты вектора состояний принимает значение «0», оно остается неизменным.
- Значение компоненты вектора состояний, на которое оказывается воздействие, в последующем остается неизменным.

Рассмотрим различные стратегии предотвращения негативных последствий от проявления дефектов в организации проекта.

Первая архитектура соответствует стратегии, ориентированной на устранение дефектов, которые могут явиться непосредственными причинами нежелательных событий (инцидентов), возникающих в ходе реализации проекта (рисунок 3.1). Широкими стрелками подчеркивается многомерность информационных потоков, циркулирующих между подсистемами.

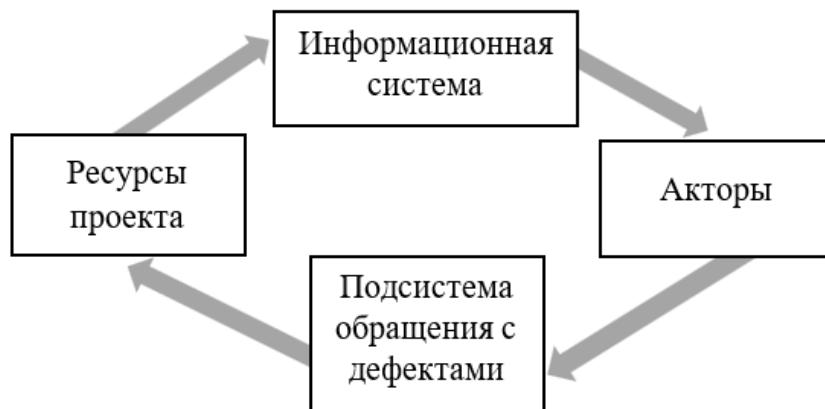


Рисунок 3.1 – Архитектура системы обращения с дефектами, ориентированная на устранение непосредственных причин инцидентов

В рамках этой архитектуры рассмотрим следующую ситуацию: Устранение непосредственных причин инцидентов (дефектов) в условиях получения объективной информации о состоянии программного продукта. Этой ситуации можно поставить в соответствие знаково-ориентированный граф, представленный на рисунке 3.2.

На рисунке 3.2 вершины графа обозначены прямоугольными блоками, подобное обозначение используется в концептуальных графах<sup>9</sup>.

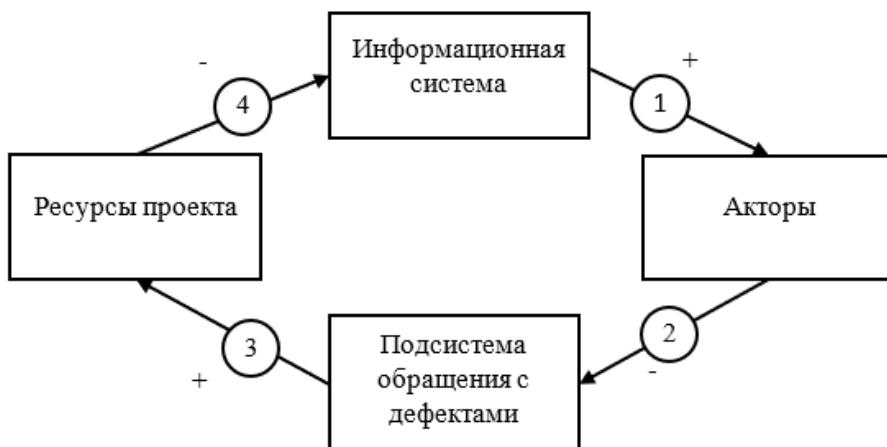


Рисунок 3.2 – Знаково-ориентированный график в условиях объективной информации о состоянии ИС

Отношения между вершинами графа (рисунок 4.2) имеют следующий смысл:

- (1) Объективная и достаточная информация о состоянии ИС положительно влияет на обоснованность решений акторов в части обеспечения функциональной надежности.
- (2) Рациональные решения акторов отрицательно влияют на количество дефектов разной природы в системе управления проектом.
- (3) Уменьшение количества дефектов в системе управления проектом отрицательно влияет на объемы ресурсов, затрачиваемых на устранение негативных последствий от проявления дефектов. Иными словами, увеличиваются объемы ресурсов, которые могут быть потрачены на обеспечение требуемых потребительских свойств ИС.
- (4) Увеличение ресурсов, выделяемых на разработку программного продукта, позволяет улучшить управление качеством промежуточных результатов проекта. Это в свою очередь позволяет сократить количество ошибок (источников латентных дефектов), совершаемых разработчиками, т.е. улучшить состояние программного продукта.

<sup>9</sup> Концептуальный граф – двудольный ориентированный граф, состоящий из вершин двух типов: понятий (*concept*) и концептуальных отношений (*conceptual relation*). Понятия в концептуальных графах отображаются прямоугольниками, отношения между ними – эллипсами

Знаково-ориентированному графу, представленному на рисунке 3.2, можно поставить в соответствие архетип «Управление с задержкой».

Данной модели, соответствует матрица  $\|H\|$  вида:

	ПП	А	Д	РП
ПП	0	1	0	0
А	0	0	-1	0
Д	0	0	0	1
РП	-1	0	0	0

ПП - программный продукт, А - акторы, Д - дефекты, РП - ресурсы проекта.

Вектор  $c_0$  представим в виде:  $\begin{vmatrix} 0 \\ \text{ПП} \end{vmatrix} \begin{vmatrix} 1 \\ \text{А} \end{vmatrix} \begin{vmatrix} 0 \\ \text{Д} \end{vmatrix} \begin{vmatrix} 0 \\ \text{РП} \end{vmatrix}$

что содержательно соответствует тому, что при реализации проекта акторы получают объективную информацию о состоянии ИС (например, в начале проекта в виде требований к продукту, оформленных согласно ГОСТ 34.601-90 [48]).

После выполнения преобразований, можно установить, что после третьего шага вычислений вектор состояний  $c_3$  приобретает вид:

$\begin{vmatrix} 1 \\ \text{ПП} \end{vmatrix} \begin{vmatrix} 1 \\ \text{А} \end{vmatrix} \begin{vmatrix} -1 \\ \text{Д} \end{vmatrix} \begin{vmatrix} -1 \\ \text{РП} \end{vmatrix}$  и более не изменяется.

Иными словами, система переходит в стабильное состояние, причем внутреннее состояние ИС улучшилось за счет сокращения количества дефектов, акторы получили новый опыт и знания, ресурсы, затрачиваемые на устранение дефектов, уменьшаются.

В рамках архитектуры (рисунок 3.1) рассмотрим иную ситуацию: Устранение непосредственных причин инцидентов (дефектов) в условиях неудовлетворительного контроля за состоянием ИС (рисунок 3.3).

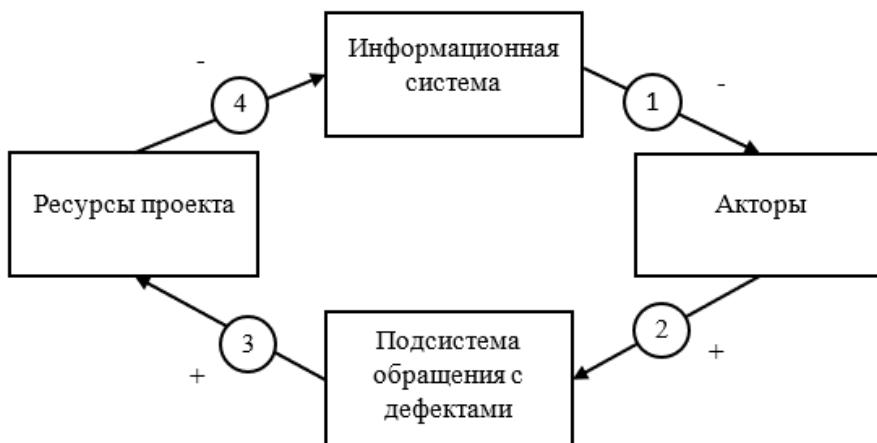


Рисунок 3.3 – Знаково-ориентированный график в условиях неудовлетворительного контроля за состоянием ИС

Отношения между вершинами графа (рисунок 4.3) имеют следующий смысл:

- (1) Необъективная и неполная информация о состоянии ИС отрицательно влияет на обоснованность решений акторов в части обеспечения функциональной надежности.
- (2) Необоснованные решения акторов приводят к увеличению дефектов разной природы в системе управления проектом.
- (3) Увеличение количества дефектов в системе управления проектом положительно сказывается на объемах ресурсов, затрачиваемых на устранение негативных последствий от проявления дефектов. Иными словами, уменьшаются объемы ресурсов, которые могут быть потрачены на обеспечение требуемых потребительских свойств ИС.
- (4) Уменьшение ресурсов, выделяемых на разработку ИС, отрицательно сказывается на контроле за качеством промежуточных результатов проекта. Иными словами, увеличивается число своевременно не выявленных ошибок, совершаемых разработчиками, что приводит к ухудшению внутреннего состояния ИС.

Данной модели, соответствует матрица  $\|H\|$  вида:

	ПП	А	Д	РП
ПП	0	-1	0	0
А	0	0	1	0
Д	0	0	0	-1
РП	1	0	0	0

Вектор  $c_0$  представим в виде: 
$$\begin{array}{c|c|c|c} 0 & 1 & 0 & 0 \\ \hline \text{ПП} & \text{А} & \text{Д} & \text{РП} \end{array}$$

что соответствует тому, что акторы получают информацию, являющуюся необъективной и недостаточной с точки зрения управления проектом (например, в начале проекта имеется в наличии документ, именуемый «Концепция информационной системы», в котором приведены слишком общие формулировки о пожеланиях заказчика, не позволяющие вынести обоснованного заключения об ожидаемых потребительских свойствах).

Далее после преобразований можно установить, что после четвертого шага вектор состояния  $c_4$  приобретает вид:

$$\begin{array}{c|c|c|c} -1 & 1 & 1 & -1 \\ \hline \text{ПП} & \text{А} & \text{Д} & \text{РП} \end{array}$$
 и более не изменяется.

Иными словами, система перешла в устойчивое состояние. Полученным результатам можно дать следующую интерпретацию: внутреннее состояние ИС (проект провален), акторы получили новые знания за счет негативного опыта, число дефектов увеличилось, ресурсы, затрачиваемые на устранение дефектов, сократились в силу того, что вложение ресурсов не приводило к повышению функциональной надежности ИС.

Представленному на рисунке 3.3 знаково-ориентированному графу можно поставить в соответствие архетип «ложные цели».

Основу рассматриваемых ниже моделей составляет стратегия «барьерного мышления» (*barrier thinking*) [200].

Стратегии, основанной на «барьерном мышлении», можно поставить в соответствие архитектуру, представленную на рисунке 3.4.

Концептуальная основа архитектуры следующая: защитные барьеры не уменьшают количества латентных дефектов в субъектоцентрической системе, но способствуют сохранению ресурсов за счет того, что препятствуют преобразованию дефектов в негативные последствия.

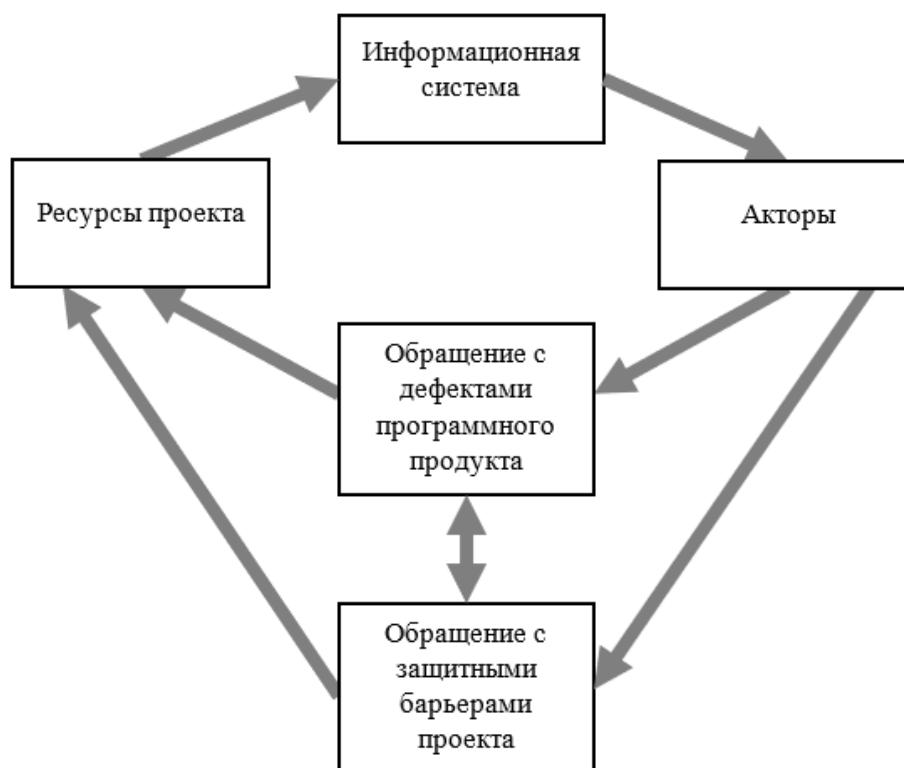


Рисунок 3.4 – Архитектура системы, реализованная в рамках стратегии «Барьерное мышление»

Рассмотрим ситуацию предотвращения дефектов за счет формирования защитных барьеров с неизменяемыми свойствами (не выполняется устранение «дыр-holes» в барьерах после их проявления). Предполагается, что акторы располагают объективной информацией о состоянии ИС.

Описанной ситуации может быть поставлена в соответствие знаково-ориентированный граф, представленный на рисунке 3.5.

Отношения между вершинами графа (рисунок 3.5) имеют следующий смысл:

- (1) Объективная и достаточная информация о состоянии ИС положительно влияет на обоснованность решений, связанных с обеспечением функциональной надежности.
- (2) Рациональные решения акторов отрицательно влияют на количество дефектов разной природы в ИС.
- (3) Рациональные решения акторов положительно влияют на выбор мест установки защитных барьеров проекта и формирования требований к их свойствам.
- (4) Уменьшение количества дефектов в проекте негативно влияет на объемы ресурсов, затрачиваемых на устранение негативных последствий от проявления дефектов. Иными словами, положительно влияет на объемы ресурсов, которые могут быть потрачены на обеспечение требуемых потребительских свойств ИС.
- (5) Использование защитных барьеров препятствует преобразованию проявлений дефектов в системе управления проектом (*hazards*) в негативные последствия (*accidents*), т.е. в дефекты в ИС. Это приводит к сокращению объемов ресурсов, затрачиваемых на устранение дефектов в ИС. Иными словами, положительно влияют на объемы ресурсов, которые могут быть потрачены на обеспечение требуемых потребительских свойств ИС.
- (6) Увеличение ресурсов, выделяемых на разработку ИС, позволяет улучшить управление качеством промежуточных результатов проекта. Иными словами, отрицательно влияет на количество ошибок (источников дефектов), совершаемых разработчиками ИС, т.е. способствует повышению функциональной надежности ИС.

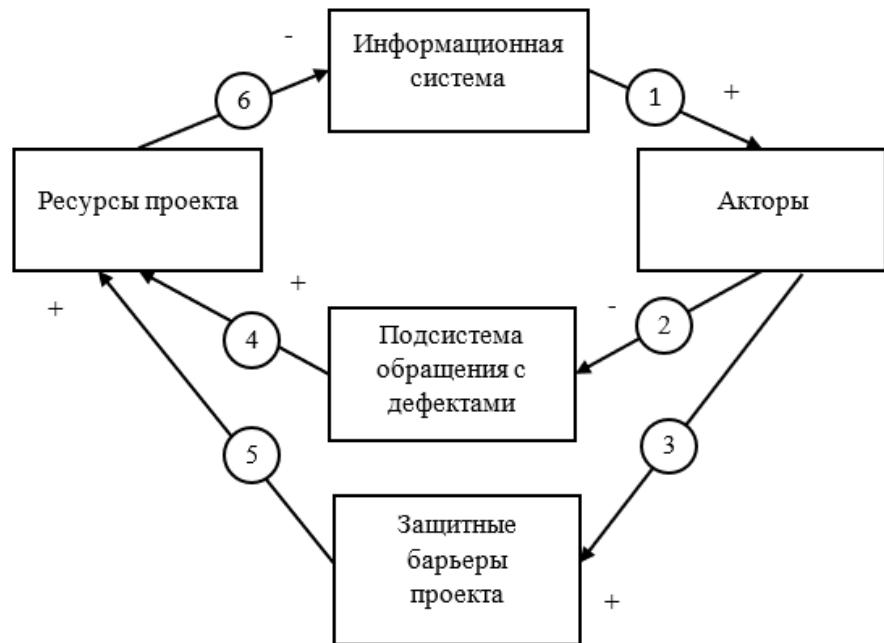


Рисунок 3.5 – Защитные барьеры с неизменными свойствами. Достоверная информация о состоянии продукта

Преобразуем граф, представленный на рисунке 3.5 к виду, представленному на рисунке 3.6, путем выделения взаимодействующих контуров (архетипное представление: архетип «Балансирующий цикл»).

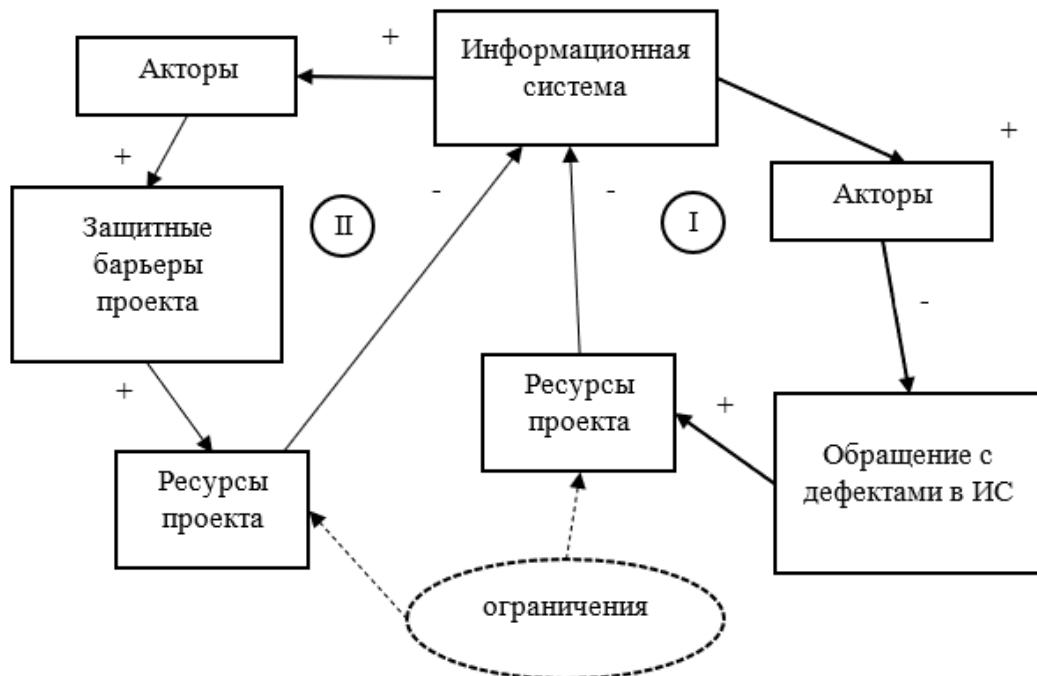


Рисунок 3.6 – Архетипное представление ситуации: Защитные барьеры с неизменными свойствами в условиях объективной информации о состоянии ИС

*Контур I:* контур положительной обратной связи. Совершенствование системы обращения с дефектами в ИС положительно сказывается на ресурсах проекта, затрачиваемых на обеспечение потребительских свойств ИС.

*Контур II:* контур отрицательной обратной связи. Смысл состоит в том, что наличие ограничений на ресурсы проекта и необходимость отвлечения ресурсов на формирование защитных барьеров проекта препятствует выделению ресурсов на совершенствование подсистемы обращения с дефектами в ИС. Графу, представленному на рисунке 3.5, ставится в соответствие матрица:

$\ H\ $ вида:	ПП	А	Д	Б	РП
ПП	0	1	0	0	0
А	0	0	-1	1	0
Д	0	0	0	0	1
Б	0	0	0	0	1
РП	-1	0	0	0	0

Вектор  $c_0$  представим в виде:

0	1	0	0	0
ПП	А	Д	Б	РП

На третьем шаге вычислений вектор состояния системы  $c_3$  приобретает вид:

0	1	0	0	0
ПП	А	Д	Б	РП

и более не изменяется.

Этому результату можно дать следующее толкование. Простая установка барьеров не улучшает внутреннего состояния ИС, а лишь блокирует проявление дефектов. Формирование барьеров не способствует сокращению ресурсов на устранение дефектов в ИС.

Рассмотрим теперь случай использования защитных барьеров в условиях недостоверной информации о состоянии программного продукта (рисунок 3.7).

Отношения между вершинами графа (рисунок 3.7) имеют следующий смысл:

- (1) Необъяснимая и недостоверная информация о состоянии ИС отрицательно влияет на обоснованность решений акторов в части обеспечения функциональной надежности.
- (2) Необоснованные решения акторов положительно влияют на количество дефектов разной природы в ИС.

- (3) Необоснованные решения акторов негативно влияют на выбор места установки защитных барьеров и формирование требований к их защитным свойствам. Иными словами, отрицательно влияют на возможности защитных барьеров проекта.
- (4) Увеличение количества дефектов в подсистеме обращения с дефектами в ИС приводит к увеличению затрат ресурсов на устранение последствий дефектов. Иными словами, негативно влияет на объемы ресурсов, выделяемых на совершенствование ценных свойств ИС с точки зрения пользователей.
- (5) Ограниченные возможности защитных барьеров в парировании дефектов, реально присутствующих в системе управления проектом, приводят к росту затрат ресурсов на их устранение. Иными словами, уменьшается объем ресурсов, которые могли бы быть направлены на развитие ценных свойств ИС с точки зрения пользователей.
- (6) Уменьшение объемов ресурсов, которые могут быть выделены на совершенствование свойств ИС, негативно сказывается на потребительских свойствах ИС.

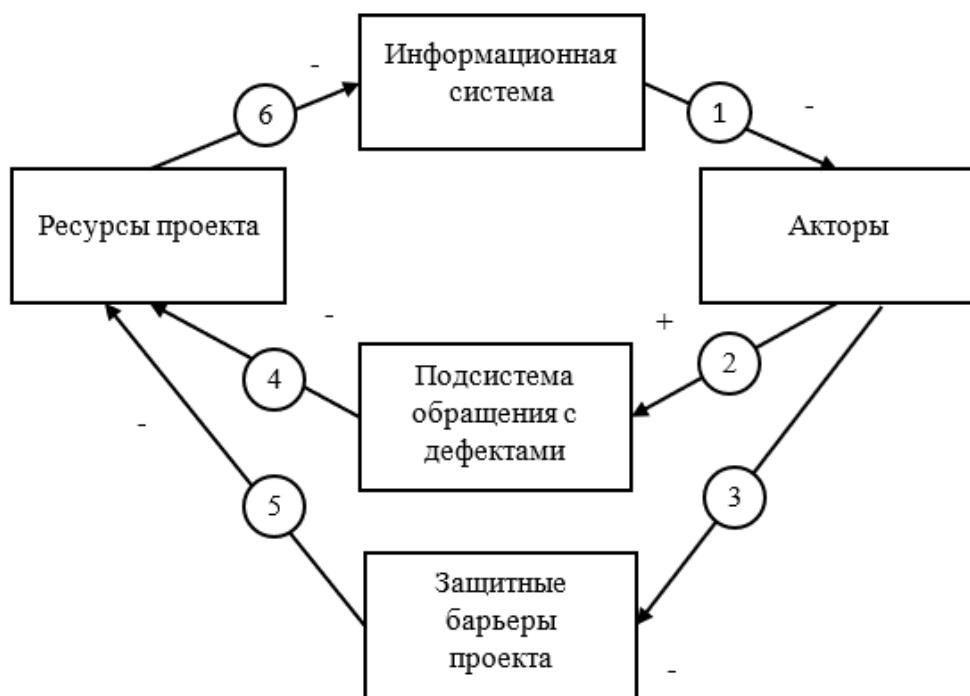


Рисунок 3.7 – Защитные барьеры с неизменными свойствами. Недостоверная информация о состоянии ИС

Преобразуем граф, представленный на рисунке 3.7 к виду, представленному на рисунке 3.8, путем выделения взаимодействующих контуров (соответствует архетипу «Трагедия общих ресурсов»).

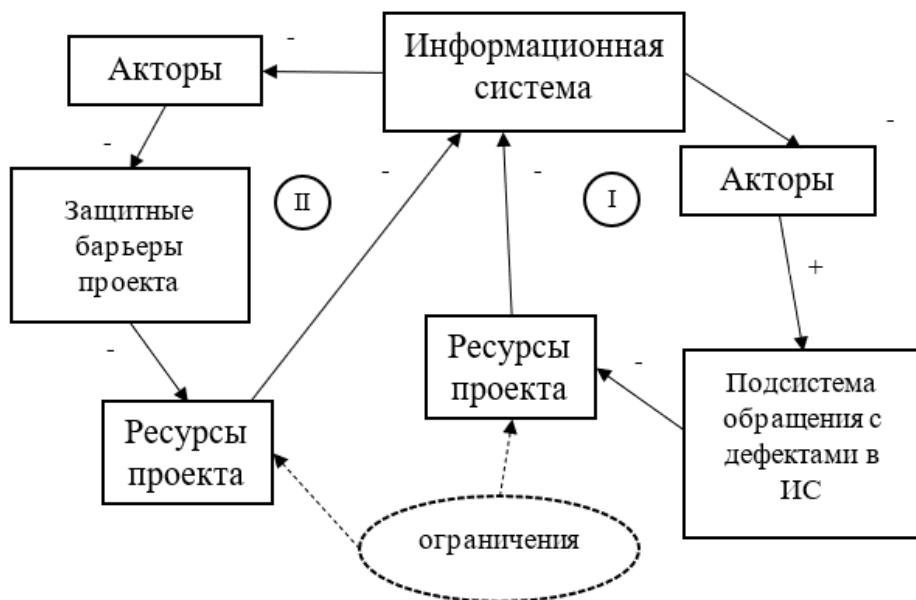


Рисунок 3.8 – Архетипное представление ситуации: Защитные барьеры с неизменными свойствами в условиях недостоверной информации

**Контур I:** Контур отрицательной обратной связи в смысле совершенствования системы обращения с дефектами в ИС. Плохая информированность акторов о состоянии ИС приводит к тому, что вне зоны внимания разработчиков оказываются существенные дефекты в ИС. При этом ограниченность ресурсов, выделяемых на изучение коренных причин дефектов (плохая информированность акторов), в том числе из-за отвлечения ресурсов на создание бесполезных барьеров, не позволяет существенно улучшить свойства ИС.

**Контур II:** Контур положительной обратной связи. Плохая информированность акторов не позволяет сформировать действенную систему защиты на основе барьеров. Это является стимулом расходования все больших ресурсов на создание защитных барьеров, однако отсутствие объективной информации о создании целевого концепта (ИС) препятствует выработке действенных решений. Отсутствие желаемого результата является стимулом для все большего вложения в создание барьеров, что оставляет все меньше ресурсов на поиск латентных дефектов.

Графу, представленному на рисунке 3.7, ставится в соответствие матрица

$\|H\|$  вида:

	ПП	А	Д	Б	РП
ПП	0	-1	0	0	0
А	0	0	1	-1	0
Д	0	0	0	0	-1
Б	0	0	0	0	-1
РП	-1	0	0	0	0

Вектор  $c_0$  представим в виде:

0	1	0	0	0
ПП	А	Д	Б	РП

На втором шаге вычислений вектор состояния системы  $c_2$  приобретает вид:

0	1	1	-1	0
ПП	А	Д	Б	РП

Это свидетельствует о переходе системы в стабильное состояние.

Полученные результаты позволяют заключить, что реализация барьерного подхода в условиях недостоверной информации не изменяет внутреннего состояния ИС, позволяет получить новые знания (хотя и на основе отрицательного опыта), благоприятно сказывается на состоянии дефектов, присутствующих в системе управления проектом, отрицательно сказывается на защитных свойствах барьеров, не изменяет ограничений на ресурсы проекта.

Далее рассмотрим случай реализации барьерного подхода с ликвидацией «дыр» в защитных барьерах в условиях достоверной информации о состоянии ИС. Этой ситуации может быть поставлена модель, представленная на рисунке 3.9.

Отношения между вершинами графа (рисунок 3.9) имеют следующий смысл:

- (1) Объективная и достаточная информация о состоянии ИС положительно влияет на обоснованность решений, связанных с обеспечением функциональной надежности.
- (2) Рациональные решения акторов отрицательно влияют на количество дефектов в ИС.
- (3) Рациональные решения акторов отрицательно влияют на количество дефектов в ИС.
- (4) Защитные барьеры отрицательно влияют на возможность проявления латентных дефектов в ИС.

- (5) Уменьшение проявлений дефектов в ИС положительно влияет на объемы ресурсов, которые могут быть потрачены на обеспечение требуемых потребительских свойств ИС.
- (6) Увеличение ресурсов, выделяемых на разработку ИС, делает возможным улучшить систему управления состоянием ИС, т.е. отрицательно влияет на количество ошибок (источников дефектов), совершаемых разработчиками ИС, что способствует повышению функциональной надежности ИС.
- (7) Исследование причин проявления дефектов в ИС (*Root cause analysis*) положительно влияет на совершенствование свойств барьеров, т.е. уменьшение дефектов («holes») в защитных барьерах.
- (8) Уменьшение дефектов в защитных барьерах положительно влияет на защитные свойства барьеров.

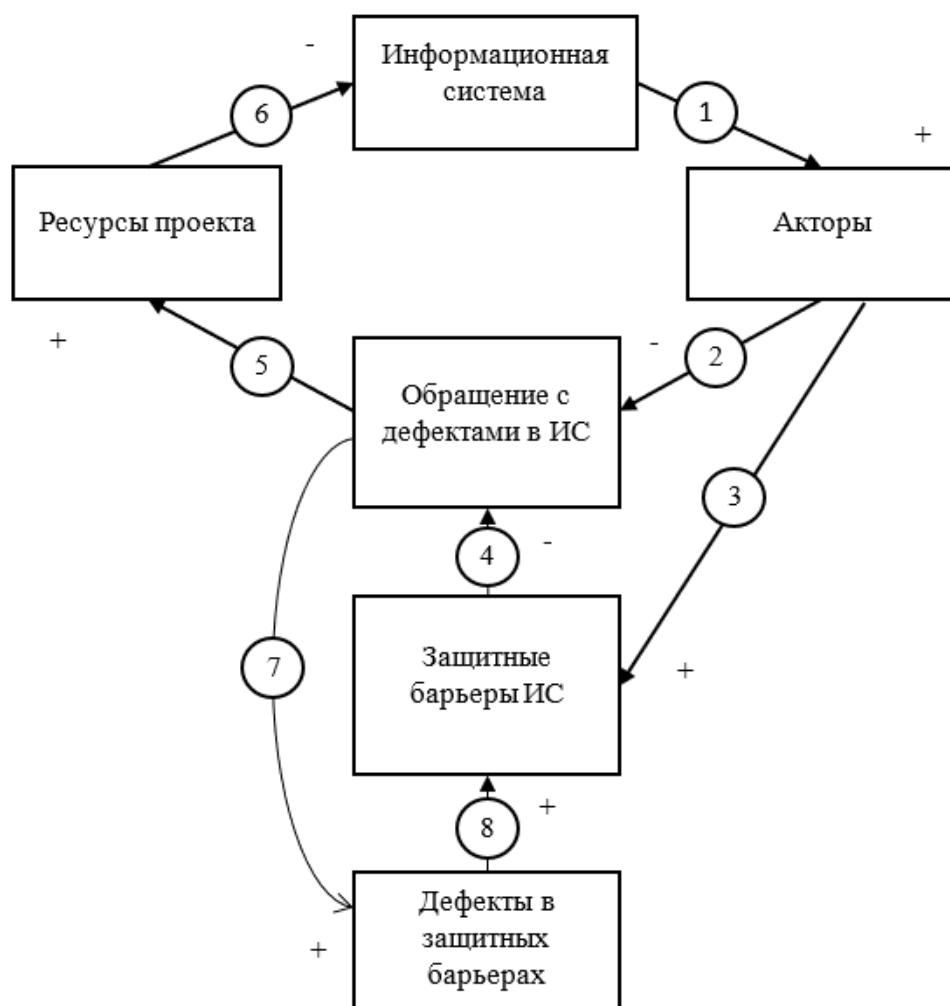


Рисунок 3.9 – Реализация «барьерного подхода» с ликвидацией «дыр» в защитных барьерах в условиях объективной информации о состоянии ИС

Путем выделения взаимодействующих контуров (архетипное представление) модель, представленная на рисунке 3.9, может быть преобразована к виду, представленному на рисунке 3.10. Этой модели соответствует архетип «Риск и недоинвестирование».

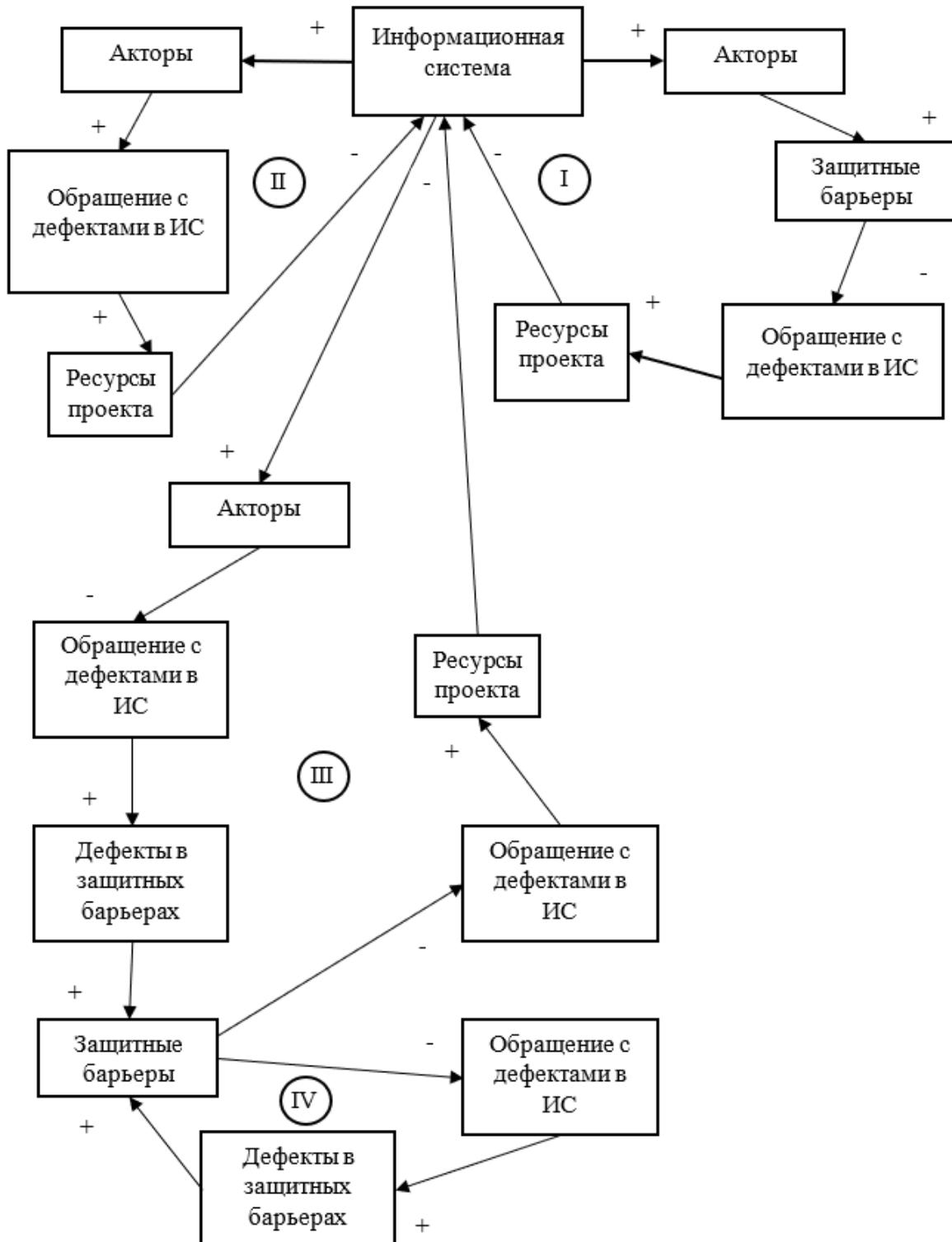


Рисунок 3.10 – Архетипное представление ситуации: Ликвидация «дыр» в защитных барьерах в условиях объективной информации о состоянии ИС

Применительно к проблеме обращения с дефектами содержание этого архетипа сводится к тому, что для обеспечения требуемых потребительских свойств ИС необходимо своевременно и в нужном объеме выделять средства не только на реализацию реактивного подхода к устранению дефектов (контур II), но и на проведение исследований, ориентированных на выявление возможных источников опасностей (латентных дефектов) и разработку мер по парированию их проявлений (контуры I, III, IV). Контуру I могут быть поставлены в соответствие подходы *AFD-1*, *AFD-2*. Контурам III и IV – подход *BowTie* [137].

Графу, представленному на рисунке 3.9, ставится в соответствие матрица:

$\|H\|$  вида:

	ПП	А	Д	РП	Б	ДБ
ПП	0	1	0	0	0	0
А	0	0	-1	0	1	0
Д	0	0	0	1	0	1
РП	-1	0	0	0	0	0
Б	0	0	-1	0	0	0
ДБ	0	0	0	0	1	0

Здесь ДБ означает «дефекты в защитных барьерах». Задав вектор  $c_0$  в виде:

$$\begin{array}{|c|c|c|c|c|c|} \hline 0 & 1 & 0 & 0 & 0 & 0 \\ \hline \text{ПП} & \text{А} & \text{Д} & \text{РП} & \text{Б} & \text{ДБ} \\ \hline \end{array}$$

получаем, что после третьего шага вектор состояний приобретает вид:

На третьем шаге вычислений вектор состояния системы  $c_3$  приобретает вид:

$$\begin{array}{|c|c|c|c|c|c|} \hline 1 & 1 & -1 & -1 & 1 & -1 \\ \hline \text{ПП} & \text{А} & \text{Д} & \text{РП} & \text{Б} & \text{ДБ} \\ \hline \end{array} \quad \text{и более не изменяется.}$$

Этому можно дать следующую интерпретацию: объективная информация о состоянии ИС, а также реализация проактивного подхода к обеспечению функциональной надежности в рамках барьерного мышления:

- улучшают состояние ИС;
- сокращают количество дефектов в ИС;
- улучшают защитные свойства барьеров; уменьшают количество дефектов в барьерах.

Рассмотрим теперь ситуацию формирования свойств защитных барьеров в условиях недостоверной информации о состоянии программного продукта. Этому случаю соответствует модель, представленная на рисунке 3.11.



Рисунок 3.11 – Реализация «барьерного подхода» с ликвидацией «дыр» в защитных барьерах в условиях недостоверной информации о состоянии ИС

Отношения между вершинами графа (рисунок 3.11) имеют следующий смысл:

- (1) Необъективная и недостаточная информация о состоянии ИС отрицательно влияет на обоснованность решений, связанных с обеспечением функциональной надежности.
- (2) Нерациональные решения акторов приводят к увеличению (положительно влияют) на количество дефектов в ИС, т.е. ухудшаются свойства системы обращения с дефектами.

- (3) Нерациональные решения акторов отрицательно влияют на выбор мест размещения защитных барьеров от проявления дефектов, имеющихся в ИС, и формировании требований к их свойствам.
- (4) Плохие свойства защитных барьеров отрицательно влияют на возможность предотвращения проявлений дефектов в ИС.
- (5) Рост проявлений дефектов отрицательно влияет на объемы ресурсов, которые могут быть потрачены на совершенствование свойств ИС из-за увеличения затрат на устранение последствий проявления дефектов.
- (6) Сокращение объемов ресурсов, выделяемых на разработку ИС, отрицательно влияет на функциональную надежность, т.к. увеличивается число совершаемых ошибок и, соответственно, латентных дефектов.
- (7) Ухудшение свойств системы обращения с дефектами не способствует объективному выявлению «дыр» в защитных барьерах.
- (8) Увеличение количества «дыр» отрицательно сказывается на защитных свойствах барьеров.

Путем выделения взаимодействующих контуров (архетипное представление) модель, представленная на рисунке 3.11, преобразована к виду, представленному на рисунке 3.12. Данной модели соответствует архетип «Трагедия общих ресурсов».

Применительно к проблеме обращения с дефектами в условиях недостоверной информации содержание архетипа «Трагедия общих ресурсов» означает, что недостаток информационных ресурсов о фактическом состоянии ИС:

- не позволяет целенаправленно воздействовать на имеющиеся дефекты (контура I, II);
- обоснованно расставлять защитные барьеры (контуру I);
- воздействовать на дефекты в защитных барьерах (контуру III).

Итогом этого является разочарование заказчика в результатах проекта.

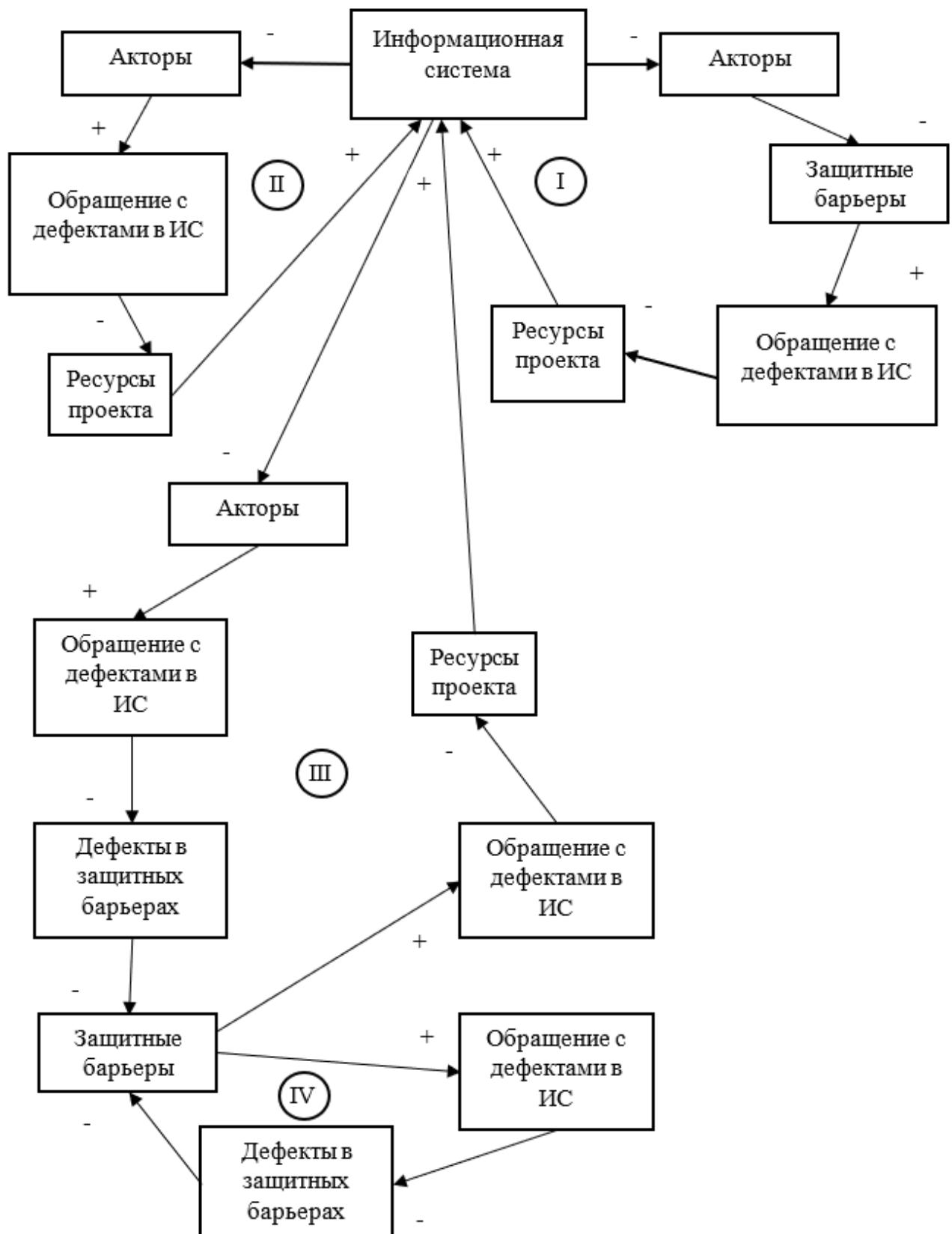


Рисунок 3.12 Архетипное представление ситуации: Ликвидация «дыр» в защитных барьерах в условиях недостоверной информации о состоянии ИС

Графу, представленному на рисунке 3.11, ставится в соответствие матрица

$\|H\|$  вида:

	ПП	А	Д	РП	Б	ДБ
ПП	0	-1	0	0	0	0
А	0	0	1	0	-1	0
Д	0	0	0	-1	0	-1
РП	1	0	0	0	0	0
Б	0	0	1	0	0	0
ДБ	0	0	0	0	-1	0

Задав вектор  $c_0$  в виде:

$$\begin{array}{|c|c|c|c|c|c|} \hline 0 & 1 & 0 & 0 & 0 & 0 \\ \hline \text{ПП} & \text{А} & \text{Д} & \text{РП} & \text{Б} & \text{ДБ} \\ \hline \end{array}$$

получаем, что после четвертого шага вектор состояний  $c_4$  приобретает вид:

$$\begin{array}{|c|c|c|c|c|c|} \hline -1 & 1 & 1 & -1 & -1 & -1 \\ \hline \text{ПП} & \text{А} & \text{Д} & \text{РП} & \text{Б} & \text{ДБ} \\ \hline \end{array} \quad \text{и более не изменяется.}$$

Результаты расчетов позволяют сделать заключение, что в условиях отсутствия объективной информации о состоянии ИС использование «барьерного подхода» не позволяет обеспечить уменьшение дефектов в ИС, несмотря на усиление защитных свойств барьеров. Это обусловлено тем, что ошибочно выбираются места размещения барьеров, а также выдвигаются ошибочные требования к их свойствам.

Таким образом, предложен подход к описанию различных архитектур подсистем обращения с дефектами посредством системных архетипов, наличие модели в виде знаково-ориентированного графа позволяет на качественном уровне посредством простого алгоритма оценить последствия от воздействий на разные узлы графа, в том числе последствия от ошибочных воздействий. Рассмотрены различные стратегии обращения с дефектами.

### **3.2 Формальная модель ошибок предпроектной стадии**

Дефекты начинают возникать на предпроектной стадии, поэтому важно распознать их источники.

На рисунке 3.13 представлена формальная модель ошибок предпроектной стадии, основные этапы предпроектной стадии рассмотрены в разделе 2.4.

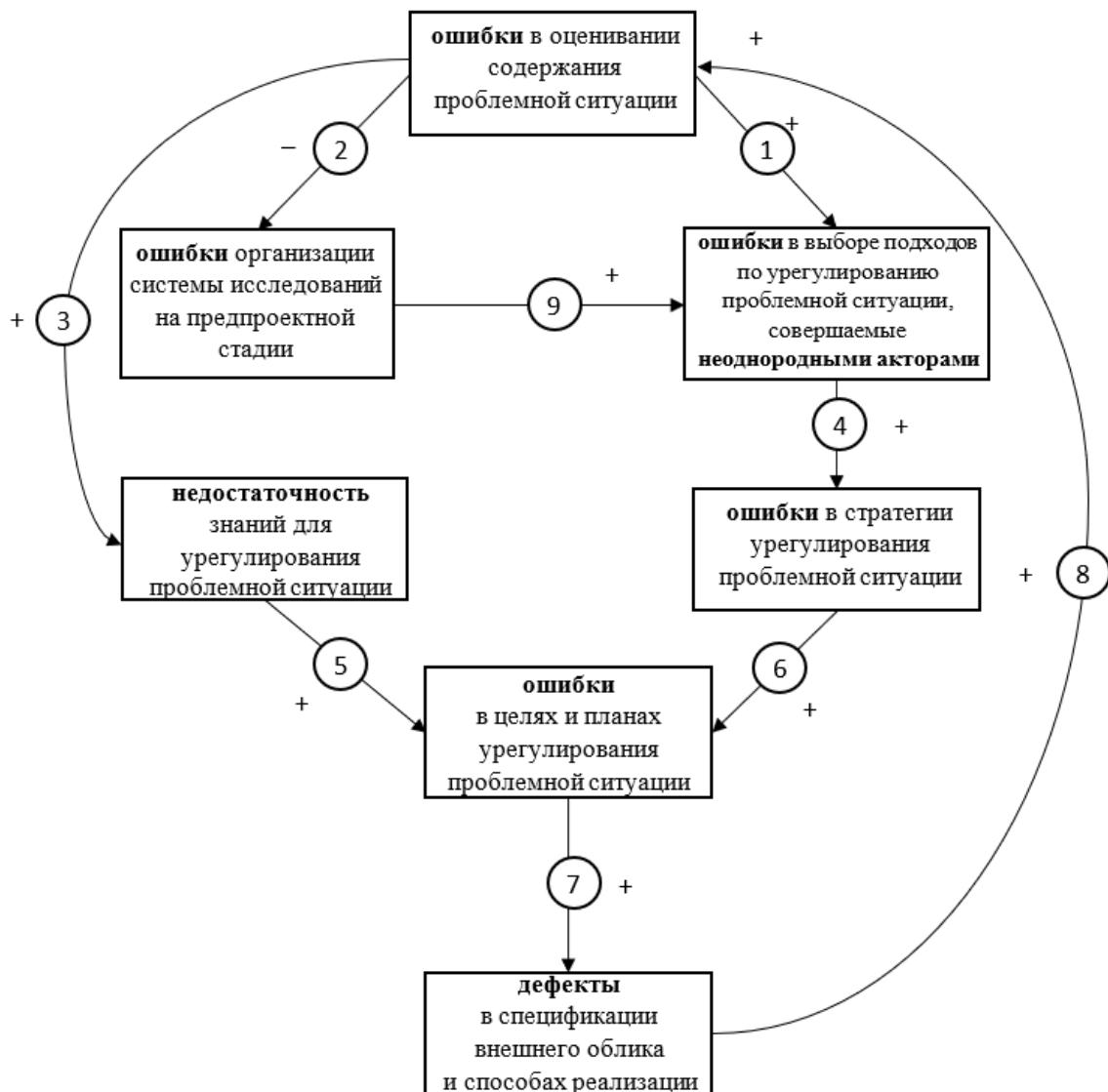


Рисунок 3.13 – Формальная модель ошибок предпроектной стадии

В таблице 3.1 приведено описание ключевых факторов и их соответствие этапам процесса предпроектной стадии (рассмотрены в разделе 2.4, рисунок 2.8). В таблице 3.2 приведено описание содержания отношений между ключевыми факторами.

Таблица 3.1 – Ключевые факторы возникновения ошибок на предпроектной стадии

<b>Ключевой фактор</b>	<b>Основные этапы процесса предпроектной стадии</b>
<i>Ошибка в оценивании содержания проблемной ситуации</i>	<ul style="list-style-type: none"> <li>– Регистрация симптомов проблемной ситуации;</li> </ul>
<i>Ошибки в выборе подходов по урегулированию проблемной ситуации, совершаемые неоднородными акторами</i>	<ul style="list-style-type: none"> <li>– Выявление неоднородных акторов, вовлеченных в урегулирование проблемной ситуации;</li> <li>– Изучение персональных онтологических и когнитивных моделей неоднородных акторов;</li> </ul>
<i>Недостаточность знаний для урегулирования проблемной ситуации</i>	<ul style="list-style-type: none"> <li>– Формирование онтологической модели и консолидированной когнитивной модели проблемной ситуации;</li> <li>– Представление проблемной ситуации в виде многосвязного объекта управления;</li> </ul>
<i>Ошибки в стратегии урегулирования проблемной ситуации</i>	<ul style="list-style-type: none"> <li>– Формирование согласованной стратегии урегулирования проблемной ситуации;</li> </ul>
<i>Ошибки в целях и планах урегулирования проблемной ситуации</i>	<ul style="list-style-type: none"> <li>– Формирование дорожной карты реализации стратегии;</li> </ul>
<i>Ошибки организации системы исследований на предпроектной стадии</i>	<ul style="list-style-type: none"> <li>– Регистрация симптомов проблемной ситуации;</li> <li>– Выявление неоднородных акторов, вовлеченных в урегулирование проблемной ситуации;</li> <li>– Изучение персональных онтологических и когнитивных моделей неоднородных акторов;</li> <li>– Формирование онтологической модели и консолидированной когнитивной модели проблемной ситуации;</li> <li>– Представление проблемной ситуации в виде многосвязного объекта управления;</li> </ul>
<i>Дефекты в спецификациях внешнего облика ИС и способах реализации</i>	<ul style="list-style-type: none"> <li>– Спецификация внешнего облика компонентов ИС.</li> </ul>

Таблица 3.2 – Описание содержания отношений между ключевыми факторами

	Содержание отношений
(1)	неверное, либо неполное понимание причин возникновения проблемной ситуации приводит к ошибкам в выборе подходов по урегулированию;
(2)	неверное понимание содержания проблемной ситуации задает неверное направление проведения предпроектных исследований;
(3)	неверное понимание проблемной ситуации приводит к неверным технологиям получения новых знаний, требуемых для урегулирования проблемной ситуации;
(4)	чем меньше понимания содержания проблемной ситуации, тем менее выбранные подходы будут способствовать её урегулированию, и тем меньший эффект даст выбранная стратегия урегулирования;
(5)	недостаточность знаний обуславливает постановку ложных целей урегулирования проблемной ситуации;
(6)	ошибки в стратегии урегулирования проблемной ситуации приводят к ошибкам в планах;
(7)	ошибки в целях и планах урегулирования проблемной ситуации приводят к дефектам в спецификации внешнего облика ИС, что, в свою очередь, является источником ошибок и дефектов в организации и выполнении проекта;
(8)	низкое качество информационного обеспечения деятельности по урегулированию проблемной ситуации является источником ошибок в оценивании содержания проблемной ситуации;
(9)	ошибки в организации исследований увеличивают количество ошибок ( <i>mistakes</i> ), совершаемых людьми.

Рассмотрим подробнее содержание контуров модели, представленной на рисунке 3.13.

Контуру, представленному на рисунке 3.14, можно дать следующее толкование: неверное понимание содержания проблемной ситуации приводит к неверной организации ее исследования (системным ошибкам). Однако низкая эффективность деятельности по урегулированию проблемной ситуации вынуждает корректировать систему исследований, что приводит к уменьшению системных ошибок.

Данный контур соответствует КОС<sub>3</sub>, выделенному в системной модели формирования понятийного пространства, стратегии и внешнего облика ИС (см. раздел 2.4, рисунок 2.10).

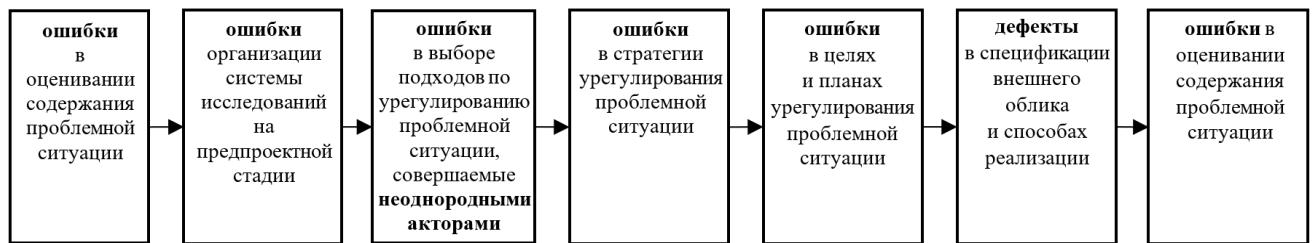


Рисунок 3.14 – Стабилизирующий контур (а)

Контру, приведенному на рисунке 3.15, можно дать следующее толкование: недостаточная эффективность деятельности по урегулированию проблемной ситуации и стремление менеджеров оправдать неудачи за счет ошибок, совершаемых исполнителями, но не просчетами, допущенными при организации системы исследований на предпроектной стадии, способствует повышению качества деятельности исполнителей, т.е. уменьшению ошибок, относящихся к классам *lapses* и *slips*. Данный контур является развивающим, соответствует КОС<sub>2</sub>, выделенному в системной модели формирования понятийного пространства, стратегии и внешнего облика ИС (см. раздел 2.4, рисунок 2.10).



Рисунок 3.15 – Развивающий контур (б)

Развивающему контру (в), представленному на рисунке 3.16, можно дать следующее толкование: невозможность урегулирования проблемной ситуации лишь за счет уменьшения ошибок, относящихся к классам *lapses* и *slips*, вынуждает получать новые знания относительно причин возникновения проблемной ситуации. Это приводит к уменьшению количества ошибок, относящихся к классу *mistakes*. Этот контур соответствует КОС<sub>1</sub>, выделенному в системной модели формирования понятийного пространства, стратегии и внешнего облика ИС (см. рисунок 2.10).



Рисунок 3.16 – Развивающий контур (в)

Предложенная модель соотносится с системным архетипом «Пределы роста».

Иными словами, возможности предупреждения возникновения ошибок и устранения дефектов лишь за счет развития технологий и инструментов имеют ограничения. Они обусловлены спецификой организации процесса исследования проблемной ситуации на основе доступных знаний, а также менталитетом субъектов, участвующих в формировании спецификации требований.

Предложенная модель позволяет сделать следующие заключения:

1) необходим баланс в распределении усилий по борьбе с ошибками разной природы, невозможно обеспечить высокое качество спецификаций только лишь за счет борьбы с ошибками, допускаемыми людьми, не совершенствуя при этом организацию системы исследования проблемной ситуации;

2) дефекты, с одной стороны, являются фактором, снижающим функциональную надежность ИС, с другой стороны, являются фактором совершенствования системы изучения проблемной ситуации за счет получения новых знаний, совершенствования методической и инструментальной базы предотвращения дефектов, их локализации и устранения.

#### *Анализ адекватности модели.*

Необходимым условием разработки формальных моделей сложных систем выступает анализ их адекватности [112].

Основу анализа адекватности структурной модели, представленной на рисунке 3.13, составляет проведение расчетов на основе таких исходных данных, для которых заранее известны результаты. Технологическую основу расчетов составляет метод, использованный в разделе 3.1.

Графу, представленному на рисунке 3.13, ставится в соответствие матрица:

$$\|H\| = \begin{array}{|ccccccc|} \hline & 0 & 1 & 1 & 0 & 0 & -1 & 0 \\ \hline & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \hline & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \hline & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \hline & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \hline & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline \end{array}$$

Рассмотрим случай воздействия с целью улучшения состояния узла «ошибки организации системы исследований на предпроектной стадии» (т.е. уменьшение ошибок в организации системы исследований на предпроектной стадии).

Этому соответствует вектор состояния вида:  $\vec{c}_0: \boxed{0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0}$

Следя расчетной схеме (3.1) на шестой итерации, получаем вектор состояния системы вида:  $\vec{c}_6: \boxed{1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1}$

На последующей итерации вектор состояний не изменяется. Это свидетельствует о переходе системы в стабильное состояние.

Интерпретируем полученный результат: Совершенствование организации системы исследований препятствует возникновению ошибок разной природы, что ускоряет урегулирование проблемной ситуации.

Аналогичным способом можно проанализировать воздействия на другие узлы графа, представленного на рисунке 3.13. Выполненный анализ результатов моделирования позволяет сделать заключение о том, что они не противоречат содержанию ситуаций, в действительности имеющих место при разработке спецификаций внешнего облика ИС.

Таким образом соответствие предложенной модели архетипу «Пределы роста», показывает, что существует предел возможностей по предупреждению дефектов, связанных как с неопределенностью проблемной ситуации, так и с компетентностью разработчиков. Это, в свою очередь, обуславливает необходимость включения в систему подготовки *IT*-специалистов дисциплин, способствующих общему развитию.

### 3.3 Модель формирования интегрального показателя, характеризующего совокупное влияние факторов, как способствующих, так и препятствующих повышению функциональной надежности информационных систем

ИС является разновидностью сложных систем, что предполагает иерархическую организацию управления свойствами: от исследования проблемной ситуации (стратегическое управление) до программирования отдельных приложений (оперативное управление).

Результат управления определяется совокупными действиями факторов, как способствующих, так и препятствующих предупреждению возникновения, выявления и устранения дефектов.

Построение формальной модели создает основу принятия рациональных решений по ослаблению влияния препятствующих фактов и усилию влияния способствующих фактов в условиях ограниченных ресурсов проекта.

Для построения математической модели интегрального показателя  $\gamma_k$ , представим системный архетип так, как показано на 3.17.

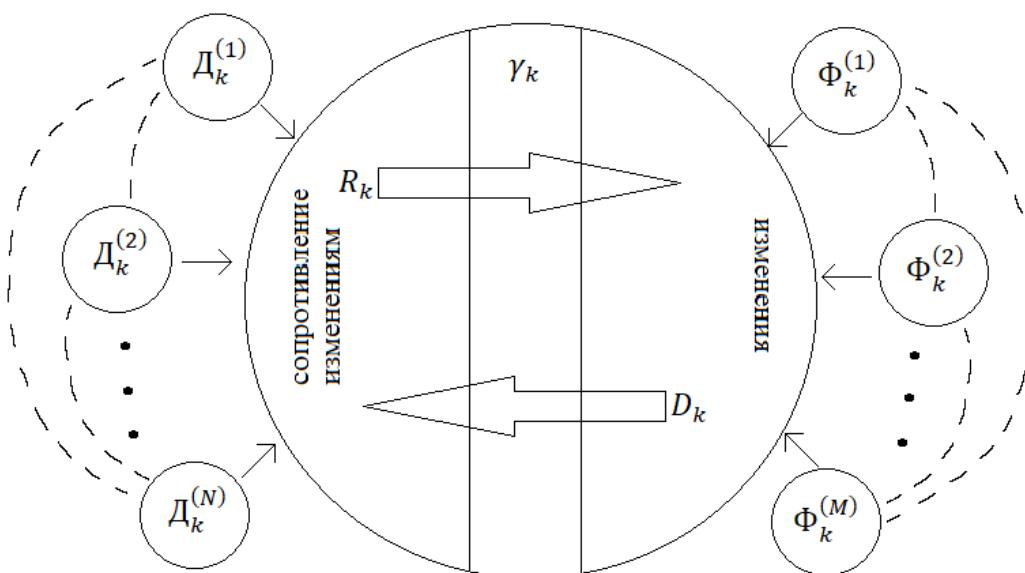


Рисунок 3.17 – Системный архетип «Пределы роста»

Факторы  $\Delta_k^{(i)}$ , ( $i = \overline{1; N}$ ) соответствуют тем, которые препятствуют устранению дефектов. Факторы  $\Phi_k^{(j)}$ , ( $j = \overline{1; M}$ ) соответствуют тем, которые способствуют устранению дефектов.

Индекс  $k$  является признаком фазы жизненного цикла ИС. Пунктирные линии обозначают взаимное влияние факторов. Стрелками  $R_k, D_k$  обозначены интегральные характеристики, соответствующие факторам, препятствующим и способствующим устранению дефектов. Линия, разделяющая круг на две части, соответствует интегральному показателю  $\gamma_k$ . Считается, что состав и значения факторов  $\{\Delta_k\}_1^N, \{\Phi_k\}_1^M$  в течение реализации проекта остаются неизменными.

Формирование интегрального показателя  $\gamma_k$  рассмотрим в рамках ресурсного подхода. Факторы  $\{\Delta_k\}_1^N$  можно рассматривать как причины расходования ресурсов на решение задач, связанных с проявлением дефектов. Факторы  $\{\Phi_k\}_1^M$  можно рассматривать как те, которые стимулируют развитие функциональных возможностей ИС, соответствующих потребностям и желаниям пользователей.

В рамках ресурсного подхода в качестве универсальной характеристики ресурсов может выступать время. Это создает основу формирования количественных характеристик факторов. Заметим, что модель, представленная на рисунке 3.17, по виду схожа с системными моделями, описанными в [1]. Отличие состоит в учете связей между факторами  $\{\Delta_k\}_1^N$  и  $\{\Phi_k\}_1^M$ .

Системной модели, представленной на рисунке 3.17, можно поставить в соответствие структурную модель «Дом качества» (*NoQ*), представленную на рисунке 3.18.

«Крыльцо» представляет собой вектор, компонентами которого являются факторы  $\{\Delta_k\}_1^N$  либо  $\{\Phi_k\}_1^M$ .

Векторам  $\|\Delta_k\|_1^N, \|\Phi_k\|_1^M$  ставятся в соответствие матрицы  $\|A^{(\Delta)}\|$ , либо  $\|A^{(\Phi)}\|$ , характеризующие взаимное влияние факторов.

«Крыше» соответствуют компоненты вектора  $\|r_k\|_1^L$  (соответствующего факторам  $\{\Delta_k\}_1^N$ ), либо компоненты вектора  $\|d_k\|_1^Q$  (соответствующего факторам  $\{\Phi_k\}_1^M$ ).

Векторы  $\|r_k\|_1^L$  и  $\|d_k\|_1^Q$  служат основой вычисления интегральных характеристик  $R_k, D_k$  соответственно.

Векторам  $\|r_k\|_1^L$  либо  $\|d_k\|_1^Q$  ставятся в соответствие квадратные матрицы  $\|C_k^{(r)}\|$  и  $\|C_k^{(d)}\|$  соответственно, характеризующие взаимное влияние компонентов векторов.

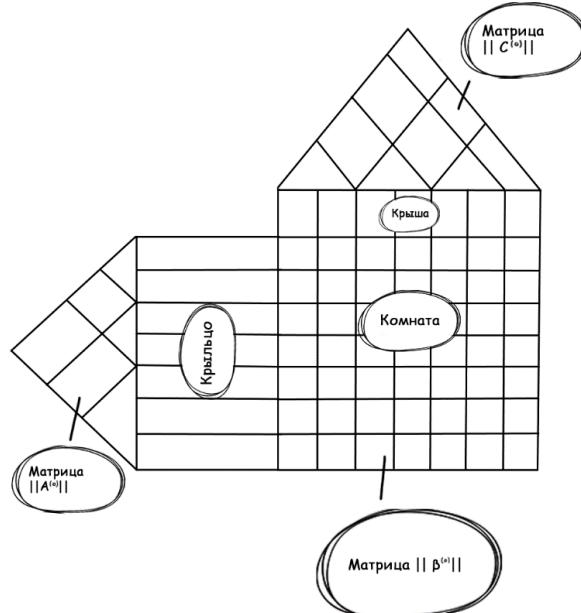


Рисунок 3.18 – Структурная модель «Дом качества»

Для преобразования квадратных матриц  $\|A_k^{(r)}\|$ ,  $\|C_k^{(d)}\|$  к виду треугольных (как показано на рисунке 3.19), воспользуемся обозначениями, показанными на том же рисунке. Диагональные элементы матриц равны нулю, что означает, что элементы векторов не могут влиять сами на себя.

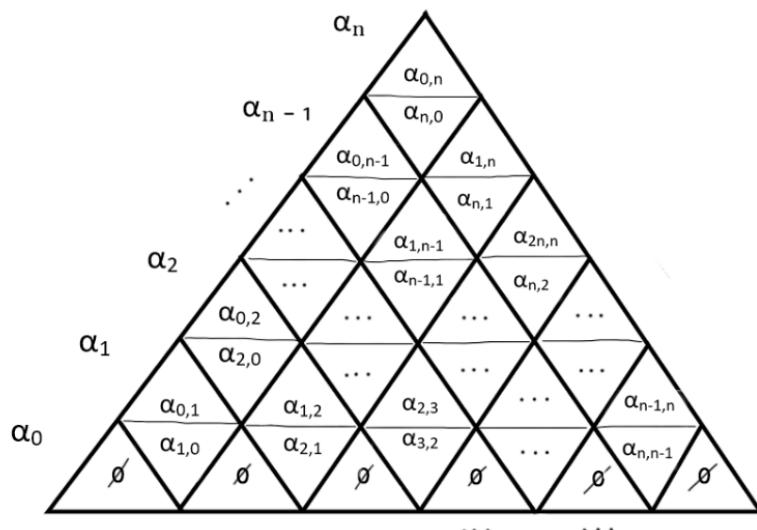


Рисунок 3.19 – Представление квадратных матриц в виде треугольных

«Комната» ставятся в соответствие прямоугольные матрицы размером либо  $N \times L$  ( $\|B_k^{(\Delta-r)}\|$  в случае, когда «крыльцу» соответствует  $\|\Delta_k\|_1^N$ , а «крыше» -  $\|r_k\|_1^L$ ), либо  $M \times Q$  ( $\|B_k^{(\Phi-d)}\|$  в случае, когда «крыльцу» соответствует  $\|\Phi_k\|_1^M$ , а «крыше» -  $\|d_k\|_1^Q$ ).

Таким образом, НоQ, соответствующие факторам  $\{\Delta_k\}_1^N$  и  $\{\Phi_k\}_1^M$ , могут быть представлены в виде трех матриц  $\|A^{(.)}\|$ ,  $\|B^{(.-.)}\|$ ,  $\|C^{(.)}\|$  и векторов факторов:

$$\begin{cases} HoQ^{(\Delta)}: (\|\Delta_k\|_1^N, \|A^{(\Delta)}\|, \|B^{(\Delta-r)}\|, \|C^{(r)}\|) \\ HoQ^{(\Phi)}: (\|\Phi_k\|_1^M, \|A^{(\Phi)}\|, \|B^{(\Phi-d)}\|, \|C^{(\Phi)}\|) \end{cases} \quad (3.2)$$

В качестве факторов  $\{\Delta_k\}_1^N$  могут выступать, например, негативное изменение ролей отдельных сотрудников в команде проекта, отсутствие устава проекта<sup>10</sup> и т.д.

В качестве компонент вектора  $\|r_k\|_1^L$ , соответствующего интегральной характеристике  $R_k$ , могут выступать, например, нарушение требований к качеству и срокам представления промежуточных и финальных результатов по задачам проекта.

Результатом воздействия факторов  $\{\Delta_k\}_1^N$  является необходимость выполнения сверхплановых работ, соответствующих каждой из компонент вектора  $\|r_k\|_1^L$ , интегральной характеристикой которых является дополнительно затраченное время.

В качестве факторов  $\{\Phi_k\}_1^M$  могут выступать, например, организационная структура проекта, основанная на уставе проекта, создание руководителем внутренней среды проекта, способствующей гармонизации личных интересов сотрудников с целями проекта [149], выявление коренных причин совершаемых ошибок и создание условий, препятствующих их повторению и другие.

Результатами воздействия факторов  $\{\Phi_k\}_1^M$  является постоянный поиск сотрудниками внутренних резервов, результатом использования которых является сокращение времени работ (без потери качества результатов), соответствующих каждой из компонент вектора  $\|d_k\|_1^Q$ .

Методическую основу формирования матриц в составе «Дома качества» составляет формальная процедура оценивания непараметрических функциональных зависимостей на основе решения обратной задачи оценивания законов распределения случайных

---

<sup>10</sup> Под уставом проекта понимается официальный документ, в котором определены зоны ответственности представителей заказчика, в соответствии с [85]

величин как функции случайного аргумента [54] с последующей аппроксимацией непараметрической модели линейной зависимостью. Использование непараметрических моделей освобождает от необходимости обоснования структуры аналитических функциональных зависимостей

В силу того, что элементам множества  $\{\Delta_k\}_1^N, \{\Phi_k\}_1^M$  на основе которых формируются интегральные характеристики  $\{r_k\}_1^L, \{d_k\}_1^Q$ , интегральных показателей  $R_k$  и  $D_k$  могут быть поставлены в соответствие сопоставимые по форме оценки одного вида – времени. Для формирования количественных оценок факторов  $\{\Delta_k\}_1^N, \{\Phi_k\}_1^M$  может использоваться подход, основанный на совокупном использовании лингвистических шкал и функций принадлежности, описанный в [194]. Модифицированный вариант этого подхода, позволяющий формализовать выбор вида функций принадлежности, описан в [67]. Ограничением указанного подхода является сложность обоснования вида функции принадлежности. В диссертации предложен подход, по сути, основанный на принципе объективности, позволяющий обосновать вид функции принадлежности. (подробно рассмотрено в разделе 4.2)

Представление факторов  $\{\Delta_k\}_1^N, \{\Phi_k\}_1^M$ , а также компонент векторов  $\|r_k\|_1^L, \|d_k\|_1^Q$ , в единицах одной размерности позволяет предложить следующую процедуру формирования компонент прямоугольных матриц, соответствующих «комнате» дома качества.

Основу расчета элементов матрицы «комната» составляет рассмотрение времени как универсальной характеристики факторов  $\{\Delta_k\}_1^N, \{\Phi_k\}_1^M$ , а также компонент интегральных характеристик  $R_k$  и  $D_k$ :  $\{r_k\}_1^L, \{d_k\}_1^Q$ .

Схема расчета имеет следующий вид:

- 1) Исторические данные о факторах  $\{\Delta_k^{(i)}\}, i \in [\overline{1, N}]; \{\Phi_k^{(j)}\}, j \in [\overline{1, M}]$ , а также  $\{r_k^{(l)}\}, l \in [\overline{1, L}]; \{d_k^{(q)}\}, q \in [\overline{1, Q}]$ , которые могут быть представлены в виде экспертных оценок различного объема, либо измерительных данных, полученных в разных условиях, преобразуются к виду однородных статистических данных.

Основу таких преобразований могут составить технологии, предложенные в [11] и подробно рассмотрены в разделе 4.1.

2) На основе полученных выборок случайных величин строятся оценки законов распределения непрерывных случайных величин:

$$\begin{aligned} A^{(\Delta)} : \left\{ \Delta_k^{(i)} \right\}_1^{N_i} &\rightarrow \hat{F}_{(k)}^{(\Delta^{(i)})}(t) \\ A^{(r)} : \left\{ r_k^{(l)} \right\}_1^{L_l} &\rightarrow \hat{F}_{(k)}^{(r^{(l)})}(t) \\ A^{(\Phi)} : \left\{ \Phi_k^{(j)} \right\}_1^{M_j} &\rightarrow \hat{F}_{(k)}^{(\Phi^{(j)})}(t) \\ A^{(d)} : \left\{ d_k^{(q)} \right\}_1^{Q_q} &\rightarrow \hat{F}_{(k)}^{(d^{(q)})}(t) \end{aligned} \quad (3.3)$$

Здесь  $A^{(\cdot)}$  - операторы, реализующие формализованную процедуру построения оценок законов распределения непрерывных случайных величин (предназначенную в том числе, для обработки выборок малого объема). Описание упомянутой процедуры приводится в [54].

3) На основе полученных оценок законов распределения непрерывной случайной величины (в качестве которой выступает время), в результате решения обратной задачи оценивания закона распределения функции случайного аргумента строятся непараметрические оценки строгих функциональных зависимостей:

$$B : \{\psi_1(\alpha), \psi_2(\beta)\} \rightarrow \beta = \varphi(\alpha), \quad (3.4)$$

Здесь  $\psi_1(\alpha), \psi_2(\beta)$  – пары оценок интегральных функций распределения;

$\beta = \varphi(\alpha)$  – строгая непараметрическая функциональная зависимость между параметрами  $\alpha$  и  $\beta$ . При необходимости эта зависимость может быть преобразована к виду параметрической.

В нашем случае в качестве параметра  $\alpha$  выступает  $\Delta_k^{(i)}, i \in [\overline{1, N}]$ ,  $\beta$  – компоненты  $r_k^{(l)}, l \in [\overline{1, L}]$ ; либо  $\Phi_k^{(j)}, j \in [\overline{1, M}]$ , и  $d_k^{(q)}, q \in [\overline{1, Q}]$ .

Каждая из функциональных зависимостей:

$$\begin{aligned} r_k^{(l)} &= \varphi_{\Delta, l}(\Delta_k^{(i)}) \\ d_k^{(q)} &= \varphi_{\Phi, q}(\Phi_k^{(j)}) \end{aligned} \quad (3.5)$$

Аппроксимируется линейной функциональной зависимостью:

$$\varphi_{\Delta, l} \left( \Delta_k^{(i)} \right) \rightarrow \hat{r}_k^{(l)} = b_k^{\left( \Delta_k^{(i)} \right), (l)} * \Delta_k^{(i)} \quad (3.6)$$

$$\varphi_{\Phi, q} \left( \Phi_k^{(j)} \right) \rightarrow \hat{d}_k^{(q)} = b_k^{\left( \Phi_k^{(j)} \right), (q)} * \Phi_k^{(j)}$$

Аналогичные преобразования могут быть выполнены к матрицам  $\|A^{(\cdot)}\|$  и  $\|C^{(\cdot)}\|$ , например:

Для «крыльца»  $HoQ$ , соответствующего факторам  $\{\Delta_k\}_1^N$ :  $\|\Delta_k\|_1^N$

$$F_k^{\left( \Delta_k^{(i_1)} \right)}(t), F_k^{\left( \Delta_k^{(i_2)} \right)}(t) \rightarrow \Delta_k^{(i_2)} = b_k^{\left( \Delta_k^{(i_1)} \right), \left( \Delta_k^{(i_2)} \right)} * \Delta_k^{(i_1)}, \quad (3.7)$$

$$i_1, i_2 \in [\overline{1, N}]; i_1 \neq i_2$$

Для «крыши» соответственно:

$$F_k^{(l_1)}(t), F_k^{(l_2)}(t) \rightarrow \hat{r}_k^{(l_2)} = b_k^{(l_1), (l_2)} * \hat{r}_k^{(l_1)}, \quad (3.8)$$

$$l_1, l_2 \in [\overline{1, L}]; l_1 \neq l_2$$

Коэффициенты  $b_k^{(\cdot), (\cdot)}$  служат элементами матриц  $\|A_k^{(\cdot)}\|$ ,  $\|B_k^{(\rightarrow)}\|$ ,  $\|C_k^{(\cdot)}\|$ .

Представление  $HoQ$  в виде (3.3) позволяет оценить с точки зрения функциональной надежности влияние изменения отдельных компонентов векторов  $\|\Delta_k\|_1^N$  и  $\|\Phi_k\|_1^M$  на компоненты векторов «крыши».

Постулируется положение о том, что одномоментно может изменяться только один из компонентов векторов, соответствующих факторам  $\{\Delta_k\}_1^N$ , либо  $\{\Phi_k\}_1^M$  и обусловленные этим изменения компонентов векторов  $\|r_k\|_1^L$  и  $\|d_k\|_1^Q$  происходят мгновенно.

Алгоритм оценки влияния  $i$ -го фактора  $\Delta_k^{(i)}$  на компоненты вектора «крыши»  $\|r_k\|_1^L$  можно представить следующим образом:

- 1) задание начальных значений компонентов векторов  $\|\Delta_k\|_1^N$ ;
- 2) внесение изменений в значение  $i$ -го компонента  $\Delta_k^{(i)} \rightarrow \Delta_k^{*(i)}$ ;
- 3) на основе значений элементов квадратной матрицы  $\|A_k^{(\Delta)}\|$ , представленной в форме, показанной на рисунке 3.19, вычисляются новые значения компонент вектора  $\Delta_k^{(\omega)} \rightarrow \Delta_k^{*(\omega)}$ ,  $\omega = \overline{1; N}$ ;  $\omega \neq i$ ;

- 4) на основе  $\|\Delta_k\|_1^N$  и матрицы  $\|B_k^{(\Delta \rightarrow r)}\|$  вычисляются значения вектора  $\left\|r_k^{*(l)}\right\|_1^L$ ;
- 5) на основе  $\left\|r_k^{*(l)}\right\|_1^L$  и квадратной матрицы  $\|C_k^{(r)}\|$ , представленной в форме, показанной на рисунке 4. 23, вычисляются значения компонент вектора  $\left\|r_k^{(l)}\right\|_1^L$ :

$$\left\|r_k^{*(l)}\right\|_1^L \rightarrow \left\|r_k^{(l)}\right\|_1^L \quad (3.9)$$

Аналогичные преобразования выполняются для «крыши» дома качества  $\|d_k\|_1^Q$  связанными с факторами  $\{\Phi_k\}_1^M$ .

Полученные значения компонент векторов  $\left\|r_k^{*(l)}\right\|_1^L$  и  $\left\|d_k^{(q)}\right\|$  служат основанием формирования интегральных показателей, характеризующих улучшения характеристики функциональной надежности  $R_k$  и препятствующих этому  $D_k$ .

Интегральный показатель функциональной надежности, формируемый в результате действия как развивающих, так и препятствующих развитию факторов, целесообразно представить в следующем виде:

$$\gamma_k = \exp(R_k - D_k) \quad (3.10)$$

Выбор такой формы интегрального показателя гарантирует его неотрицательность при любых  $R_k, D_k$ .

Интерпретация показателя:

$\gamma_k < 1$ , соответствует ситуации, когда сопротивление развитию превышает стремление к развитию (т.е.  $R_k$  превышает  $D_k$ );

$\gamma_k > 1$ , соответствует ситуации, когда стремление к развитию превышает сопротивление развитию (т.е.  $R_k$  меньше  $D_k$ );

$\gamma_k = 1$ , соответствует ситуации, когда неизменное состояние функциональной надежности ИС.

Таким образом, предложен формальный подход к формированию интегрального показателя, характеризующего совокупное влияние факторов как способствующих, так и препятствующих повышению функциональной надежности ИС, основу которого составляет модель «Дом качества».

В рамках ресурсного подхода предложен способ представления в сопоставимой форме закона распределения временных затрат на реализацию/парирование действия различных факторов, влияющих на состояние объекта управления. В целом, предложено новое системное сочетание известных в системном анализе моделей и подходов, что позволяет повысить степень формализации исследований, ориентированных на оценку влияния качества управления проектом на функциональную надежность ИС.

### **3.4 Динамические модели проблемных ситуаций на основе системных архетипов**

В ряде руководств по управлению проектами, например [126, 127], указывается, что функциональная надёжность ИС в значительной степени зависит от качества организации реализации проекта. Системные архетипы являются концентрированной формой представления схожих по содержанию проблемных ситуаций, встречающихся при управлении сложными системами разной природы, учитывая, что проекты ИС относятся к классу сложных систем. ИС относятся к классу сложных систем в силу того, что их неотъемлемым свойством является неопределенность внутреннего состояния.

Результаты, полученные на основе структуризации проблемных ситуаций, формируют основу информационной поддержки задачи обеспечения функциональной надежности ИС, а также создают основу для построения в различной степени формализованных знаковых моделей.

Представление проблемных ситуаций, связанных с обеспечением функциональной надежности, в виде контурных моделей позволяет перейти от качественного описания ситуаций к математическим моделям, позволяющим получить количественные оценки отдельных ситуаций.

Анализ типовых ситуаций, связанных с обеспечением функциональной надежности ИС выполним на основе архетипа «Пределы роста» (содержание архетипа подробно рассмотрено в разделе 2.5, графическая модель представлена на рисунке 2.12).

Контурная модель проблемной ситуации, связанной с обеспечением функциональной надежности ИС, соответствующая архетипу «Пределы роста» показана на рисунке 3.20.

В соответствии с контурным представлением поведение системы определяется сложным взаимодействием усиливающих и ослабляющих контуров, возникающих в системах с обратными связями.

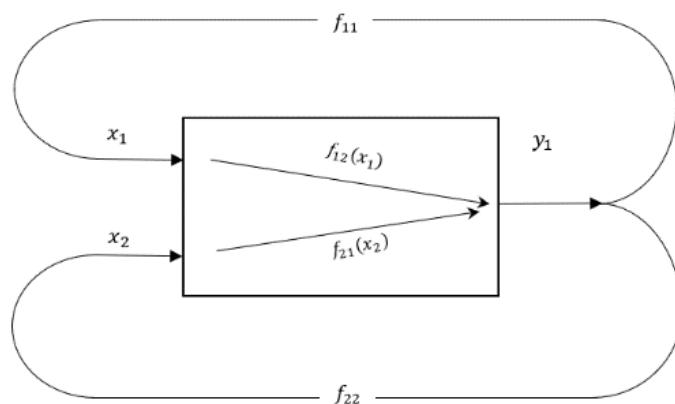


Рисунок 3.20 – Контурная модель, соответствующая архетипу «Пределы роста»

Входными параметрами модели являются:

$x_1$  - факторы, стимулирующие повышение функциональной надежности ИС;

$x_2$  - факторы, препятствующие повышению функциональной надежности ИС;

Выходным параметром является:

$y_1$  – оценка уровня функциональной надежности, т.е. эмерджентные свойства, возникающие в результате сложных взаимодействий стимулирующих и препятствующих факторов.

Содержание архетипа «Пределы роста» применительно к задачам обеспечения функциональной надежности, сводится к тому, что, с одной стороны, имеет место стремление развивать и совершенствовать потребительские свойства ИС, в том числе за счет предотвращения негативных последствий от проявления

латентных дефектов (развивающий цикл, действия ведущие к улучшению). С другой стороны, в силу того что устранение и предотвращение возникновения дефектов требует использования ресурсов проекта (при том, что не добавляет новых функциональных возможностей ИС), возникает противодействие расходованию ресурсов на решение задач обеспечения функциональной надежности (стабилизирующий цикл, условия, препятствующие улучшению).

Системному архетипу «Предел роста» как инструменту анализа проблемных ситуаций, связанных с обеспечением функциональной надежности, можно поставить в соответствие математическую модель, предложенную В.И.Арнольдом, именуемую в первоисточнике логистической, представленную в [97].

$$h_k^{(t+1)} = h_k^{(t)} \gamma_k (1 - h_k^{(t)}) \quad (3.11)$$

Параметрам этой модели с точки зрения функциональной надежности ИС можно дать следующее толкование:

$h_k^{(t)}$  – интегральный показатель функциональной надежности на  $k$ -й стадии жизненного цикла ИС.

Функциональная надежность характеризуется долей функциональных компонентов ИС, поведение которых удовлетворяет требованиям пользователей, совпадает с содержанием документации, причем в случае проявления латентных дефектов они устраняются за такое время, что отказы могут рассматриваться как сбои.

Значение  $h_k^{(t)}=0$  означает, что в системе нет ни одной работоспособной функции, т.е. такая система для пользователя интереса не представляет. При росте  $h_k^{(t)}$  функциональные возможности ИС все более соответствуют требованиям пользователей. Очевидно, что  $0 < h_k^{(t)} < 1$ , что содержательно соответствует тому, что для пользователя представляют интерес лишь системы, содержащие работоспособные функции.

$\gamma_k$  - интегральный показатель, характеризующий совокупное влияние факторов, как способствующих, так и препятствующих повышению

функциональной надежности. Считается, что в ходе реализации проекта создания ИС эта характеристика остается неизменной.

Зависимость (3.11) создает основу для исследования влияния соотношения показателей состояния ИС  $h_k^{(t)}$  и организации проекта  $\gamma_k$ .

Отправной точкой исследования является определение состояния равновесия ИС, т.е. такого, при котором показатель состояния не изменяется во времени.

В этом случае  $h_k^{(t+1)} = h_k^{(t)}$  при любом  $t$ .

При этом отношение (3.11) преобразуется к виду (3.12):

$$\gamma_k^* = \frac{1}{1-h_k^{(t)}} \quad (3.12)$$

Соотношение (3.12) соответствует случаю, когда влияние факторов, способствующих и препятствующих повышению функциональной надежности одинаково.

Выясним как изменяется показатель  $h_k^{(t)}$  в случае, когда  $\gamma_k > \gamma_k^*$ .

На рисунке 3.21 представлены результаты, соответствующие разным  $\gamma_k$  и различным  $h_k^{(0)}$  в случае, когда  $\gamma_k = 2$ , что соответствует  $h_k^{(0)} = 0.5$ .

На рисунке 3.22 представлены результаты, демонстрирующие изменение  $h_k^{(t)}$  в случае, когда  $\gamma_k = 2$ , но  $h_k^{(0)} = 0.1$ , чему соответствует  $\gamma_k^* = 9/10$ .

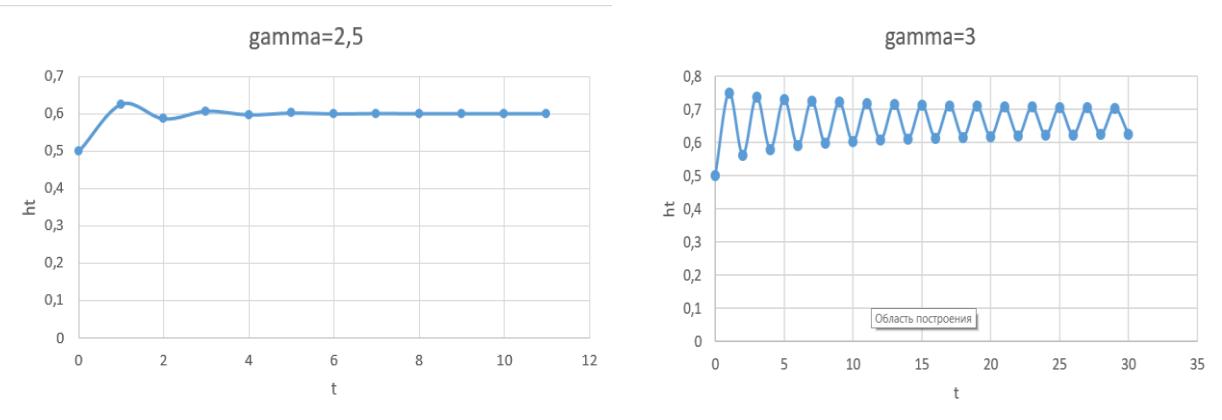


Рисунок 3.21 –Характер изменения  $h_k^{(t)}$  при  $\gamma_k > \gamma_k^*$  при  $h_k^{(0)} = 0.5$

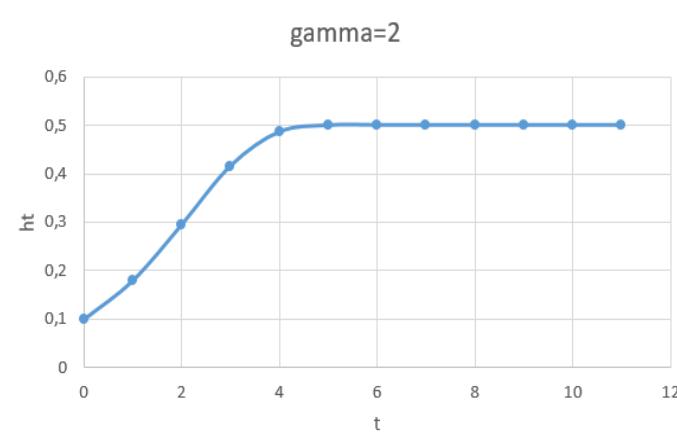


Рисунок 3.22 – Характер изменения  $h_k^{(t)}$  при  $\gamma_k > \gamma_k^*$  при  $h_k^{(0)} = 0,1$

На рисунке 3.23 представлены результаты, соответствующие случаю  $\gamma_k < \gamma_k^*$  при том же  $h_k^{(0)} = 0,5$ . На рисунке 3.24 представлены результаты, демонстрирующие изменение  $h_k^{(t)}$  в случае, когда  $\gamma_k = 2$ , но  $h_k^{(0)} = 0,9$ , чему соответствует  $\gamma_k^* = 10$ .

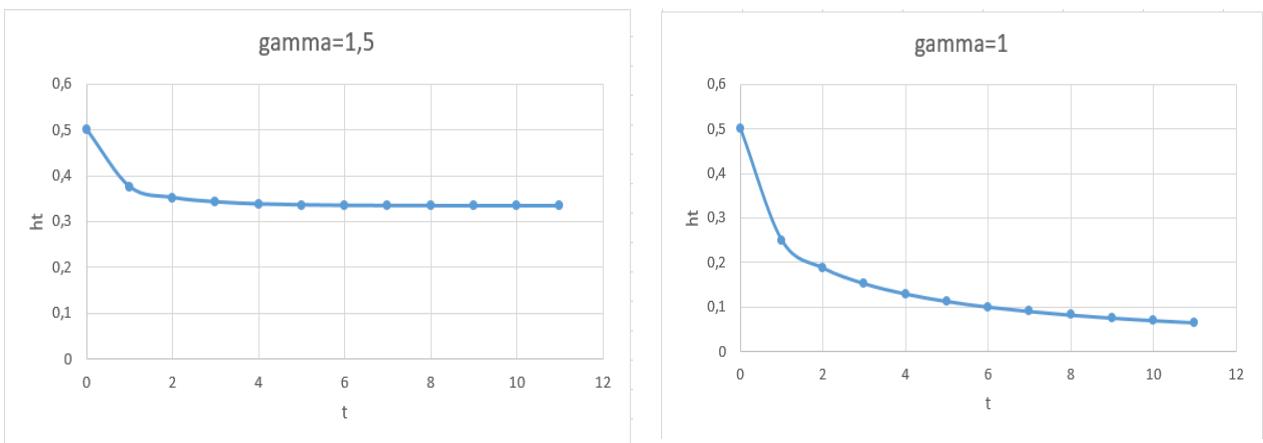


Рисунок 3.23 –Характер изменения  $h_k^{(t)}$  при  $\gamma_k < \gamma_k^*$ ,  $h_k^{(0)} = 0,5$

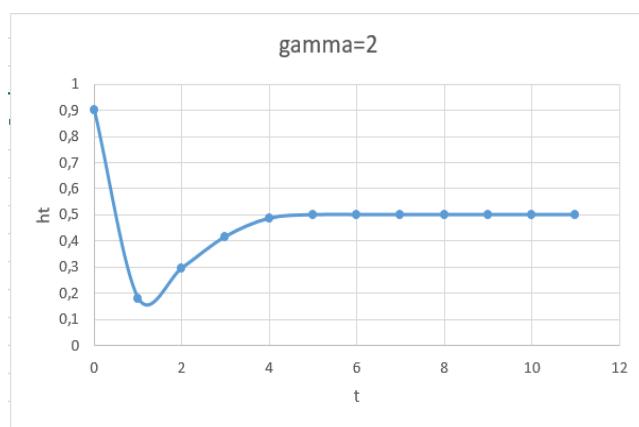


Рисунок 3.24 – Характер изменения  $h_k^{(t)}$  при  $\gamma_k < \gamma_k^*$  при  $h_k^{(0)} = 0,9$

Таким образом, на примере системного архетипа «Пределы роста» предложен формальный подход к анализу соответствия качества управления проектом создания ИС состоянию функциональной надежности ИС.

Выполнены оценки изменения состояния ИС при различных соотношениях состояния объекта управления и интегрального показателя, характеризующего качество управления проектом посредством оценки совокупного влияния развивающих и противодействующих развитию факторов. Основу оценивания составила модель, в литературе именуемая «логистической». Результаты моделирования могут быть использованы как основа анализа альтернатив обеспечения функциональной надежности ИС (формальной характеристикой альтернатив является зависимость  $\gamma_t$ ).

### **Выводы по главе 3**

1. Предложен подход к описанию различных архитектур подсистем обращения с дефектами посредством системных архетипов. Наличие модели в виде знаково-ориентированного графа позволяет на качественном уровне посредством простого алгоритма оценить последствия от воздействий на разные узлы графа, в том числе последствия от ошибочных воздействий.

2. Разработана формальная модель ошибок предпроектной стадии. Установлено, что предложенной модели может быть поставлен в соответствие архетип «Пределы роста», из чего следует, что существует предел возможностей по предупреждению дефектов, обусловленный как неопределенностью проблемной ситуации, так и компетентностью разработчиков. Предлагаемый подход отличается тем, что источники ошибок и дефектов выявляются на предпроектной стадии, в отличие от известных подходов, которые ориентированы на выявление дефектов в условиях, когда уже имеется техническое задание.

3. В рамках ресурсного подхода предложена модель формирования интегрального показателя, характеризующего совокупное влияние факторов, как способствующих, так и препятствующих повышению функциональной надежности. Основу модели составляет системное сочетание модели «Дом качества» и системных архетипов. Это позволяет повысить степень формализации исследований, ориентированных на оценку влияния качества управления проектом на функциональную надежность ИС.

4. Разработаны контурные и динамические модели проблемных ситуаций, связанных с обеспечением функциональной надежности на основе системных архетипов, весовые характеристики которых определяются математико-статистическими методами, это позволяет перейти от качественного описания ситуаций к математическим моделям, позволяющим получить количественные оценки отдельных ситуаций.

Основные результаты, представленные в третьей главе, опубликованы в статьях в рецензируемых печатных изданиях из перечня, утвержденного ВАК [16, 19, 23, 24] и в международных изданиях, индексируемых SCOPUS [135].

## ГЛАВА 4. МЕТОДИЧЕСКИЕ ОСНОВЫ ОБЕСПЕЧЕНИЯ ФУНКЦИОНАЛЬНОЙ НАДЕЖНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ

Данная глава посвящена методическим основам обеспечения функциональной надежности ИС на предпроектной стадии.

Первый параграф настоящей главы посвящен разработке метода построения многомерных зависимостей на основе совместного использования измерительных данных и экспертных оценок. Основная идея метода заключается в приведении экспертных оценок и измерительных данных к единой форме представления – закону распределения непрерывной случайной величины, что обеспечивает их сопоставимость.

Второй параграф посвящен разработке метода оценки состояния функциональной надежности информационных сервисов с учетом субъективных оценок пользователей на основе концепции профилей.

Третий параграф посвящен разработке метода структурного моделирования отказов функциональных компонент ИС, основанного на совместном использовании аппарата схем сопряжения и таблиц истинности. Данный подход создает основу для построения структурных моделей, обеспечивающих моделирование отказов в рамках как нисходящего, так и восходящего подходов к исследованию.

В четвертом параграфе с целью разноспектрного анализа функциональной надежности ИС на предпроектной стадии предложена схема анализа событий в рамках диверсионного анализа. Данный подход создает основу для анализа процессов разработки ИС с точки зрения выявления мест возможных появлений дефектов и является реализацией базового положения стандарта ESA-PSS-05-10<sup>11</sup> «Идеально реализованный процесс производства гарантирует идеальное качество продукта».

---

<sup>11</sup> ESA-PSS-05-10 Guide to software verification and validation. Prepared by: ESA Board for Software Standardisation and Control (BSSC), 1995

#### **4.1 Метод построения многомерных зависимостей на основе совместного использования измерительных данных и экспертных оценок**

Использование сведений о причинах и местах возникновения дефектов и обусловленных этим негативных последствий в настоящее время приобретает особую значимость в силу высокой цены негативных последствий, связанных с недостаточной функциональной надежностью ИС, примеры последствий представлены в [2, 210].

Сложность организации как процессов производства, так и устройства с одной стороны, стремление к регламентации разработки с другой (обоснованием к такому утверждению служит стандарт *CMMI*<sup>12</sup>) является причиной того, что на практике для разноспектрного исследования свойств ИС, включая функциональную надежность, получили развитие эмпирические и полуэмпирические модели.

Одним из аспектов концептуальных основ обеспечения функциональной надежности выделено следующее:

- основу обеспечения функциональной надежности составляет комплексное использование сведений, получаемых из разных источников: структурный анализ архитектур систем обеспечения функциональной надежности и внутреннего устройства ИС, результаты обработки исторических данных (включая метрические характеристики), связанные с проявлением дефектов разной природы, экспертные оценки субъектов, причастных к созданию и использованию ИС, составляет основу системного подхода к обеспечению функциональной надежности на предпроектной стадии.

---

<sup>12</sup> *CMMI* (*Capability Maturity Model Integration*) модель совершенствования процессов и улучшения показателей деятельности в организациях разных размеров и видов деятельности, модель разработана для систематизации и оптимизации жизненного цикла разработки программных систем. Стандарт фокусируется на повышении зрелости процессов, минимизации рисков и обеспечении предсказуемости результатов за счет внедрения структурированных практик [124].

Классический подход к решению данной задачи основан на использовании эмпирических (регрессионных), характеризующих связь параметров условий и процессов реализации проектов создания ИС и параметров, характеризующих свойства полученных ИС. Традиционные подходы к построению регрессионных зависимостей [57] предполагают наличие таблицы совместно наблюдаемых значений независимой и зависимой случайных величин.

На практике формирование таблицы совместно наблюдаемых значений может встречать определённые трудности. Это обусловлено, например, тем, что сбор различных данных осуществляется несколькими организациями, деятельность которых регламентируется разными документами, либо недостатками организации и реализации процессов формирования и обеспечения сохранности архивных данных. Либо вследствие того, что для оценки параметров состояния сложных систем использовались мнения экспертов.

Регрессионные зависимости являются одним из основных инструментов построения эмпирических дескриптивных моделей безынерционных объектов.

К настоящему времени развит теоретический аппарат построения регрессионных моделей, на основе которого разработаны программно-реализованные инструментальные средства. При этом особое место в задачах регрессионного анализа занимают линейные модели. Это обусловлено тем, что они создают основу изучения основных свойств объектов в условиях малого числа измерительных данных, что характерно при исследовании процессов производства и потребительских свойств ИС в силу уникальности проектов [4, 32, 83].

Информационную основу построения регрессионных зависимостей составляет таблица совместно наблюдаемых значений независимых и зависимых случайных величин. При решении практических задач приходится сталкиваться с ситуацией, когда формирование таких таблиц встречает ряд трудностей. В таких условиях недостаток измерительных данных приходится компенсировать экспертными оценками разного вида: в виде ожидаемых значений; в виде интервала возможных значений, в виде совокупности ожидаемых значений и интервала возможных значений случайной величины.

В силу отмеченных обстоятельств представляет интерес разработка метода преобразования измерительных данных и экспертных оценок, задаваемых в разных формах к виду таблицы совместно наблюдаемых значений, что послужит основой построения многомерных регрессионных зависимостей.

Предлагается подход к построению многомерных регрессионных зависимостей в случае, когда исходные данные, соответствующие разным компонентам векторов независимых и зависимых переменных, представляются либо в виде результатов измерений, либо в форме различных экспертных оценок.

Классическому подходу к построению регрессионных зависимостей можно поставить в соответствие схему:

$$A^{(0)}: \{\vec{x}, \vec{y}\}_1^N \rightarrow \vec{y} = \varphi_0(\vec{x}, \vec{\Theta}) \quad (4.1)$$

Здесь  $\{\vec{x}, \vec{y}\}_1^N$  - множество совместно наблюдаемых значений компонент векторов независимой  $\vec{x}$  и зависимой  $\vec{y}$  величин;

$\varphi_0(\vec{x}, \vec{\Theta})$  – функциональная зависимость, задаваемая в параметрической форме;  $N$  – число пар значений  $\vec{x}$  и  $\vec{y}$ .

В литературе описаны различные операторы  $A^{(0)}$ , например, реализующие метод «наименьших квадратов» [72], либо «метод наименьших модулей» [89].

При решении практических задач, вследствие сложности организации сбора исходных данных, формирование  $\{\vec{x}, \vec{y}\}_1^N$  встречает серьезные трудности.

В работе [54] описан подход к построению одномерных непараметрических строгих функциональных зависимостей на основе решения обратной задачи построения закона распределения функции случайного аргумента (прямая задача описана в [96]).

Схема решения обратной задачи имеет вид:

$$A^{(1)}: \{F(x), F(y)\} \rightarrow y = \varphi_1(x) \quad (4.2)$$

где  $F(x)$ ,  $F(y)$  – оценки одномерных законов распределения независимой и зависимой случайных величин, определяемые на основе обработки выборочных данных.

$$A^{(2)}: \{x\}_1^N \rightarrow x = F(x), A^{(2)}: \{y\}_1^M \rightarrow y = F(y) \quad (4.3)$$

Особенностью схемы (4.2) является то, что свойства выборочных данных  $\{x\}_1^N, \{y\}_1^M$  (объем, точность регистрации) могут быть различными. Ограничением схемы (4.2) является необходимость обоснования самого факта наличия строгих зависимостей, например, исходя из физического содержания задачи.

В работе [96] приведены результаты исследования свойств оценок строгих функциональных зависимостей  $\varphi_1(x)$ , получаемых посредством (4.2). Известно большое количество работ, ориентированных на построение оценок закона распределения случайной величины по выборкам малого объема, например [33, 61, 109]. На практике, как правило, широко используют нормальный и показательный законы распределения случайных величин.

Экспертные оценки представляются в виде оценки ожидаемого значения случайной величины  $M[z]$  (иными словами, оценки математического ожидания) и/или интервала возможных значений случайной величины  $Z$  (возможны следующие варианты представления интервалов:  $z \in [a_z, b_z]; z \in [a_z, \infty]$ ).

В работах, ссылка на которые давалась выше, рассмотрены следующие частные задачи построения оценок законов распределения случайных величин на основе экспертных оценок:

- если известны лишь границы интервала возможных значений случайной величины, т.е.  $z \in [a_z, b_z]$ , то:

$$A^{(3)}: [a_z, b_z] \rightarrow F^{(3)}(z) \quad (4.4)$$

В этом случае оптимальной оценкой  $F^{(3)}(z)$  является равномерный закон распределения;

- если известны математическое ожидание  $M[z]$ , а интервал возможных значений случайной величины представляется в виде  $[a_z, \infty]$ , то построение оценки сводится к решению задачи:

$$A^{(4)}: \{M[z], [a_z, \infty)\} \rightarrow F^{(4)}(z) \quad (4.5)$$

Оптимальной оценкой  $F^{(4)}(z)$  в этом случае является показательный закон распределения:

$$F^{(4)}(z) = 1 - e^{-\lambda z} \quad (4.6)$$

Параметр закона распределения определяется соотношением  
 $\lambda = (M[z] - a_z)^{-1}$ .

– Если известны  $M[z]$  и границы интервала  $[a_z, b_z]$ , то оценка закона распределения ищется в результате решения задачи:

$$A^{(5)} : \{M(z), [a_z, b_z]\} \rightarrow F^{(5)}(z) \quad (4.7)$$

Оценка представляется в виде:

$$F^{(5)}(z) = \int_{a_z}^z e^{\mu_0 + \mu_1 \tau} d\tau \quad (4.8)$$

причем параметры модели  $\mu_0, \mu_1$  находятся в результате решения системы уравнений:

$$\begin{cases} \int_{a_z}^{b_z} e^{\mu_0 + \mu_1 \tau} d\tau = 1 \\ \int_{a_z}^{b_z} \tau e^{\mu_0 + \mu_1 \tau} d\tau = M[z] \end{cases} \quad (4.9)$$

В частном случае, если дополнительно известно, что  $F^{(5)}(a_z) = 0, F^{(5)}(b_z) = 1, F^{(5)}(z)$  представляет собой треугольное распределение.

Преобразование экспертных оценок к виду законов распределения непрерывных случайных величин делает возможным расширить схему (4.2) на случай построения одномерных непараметрических регрессионных зависимостей на основе совместного использования измерительных данных и экспертных оценок. Вместе с тем, методы построения строгих функциональных зависимостей на основе решения обратной задачи построения закона распределения функции случайного аргумента, в том числе на основе совместного использования измерительных данных и экспертных оценок, ориентированы на исследование одномерных зависимостей и не адаптированы к исследованию поведения многомерных систем.

Далее описывается подход, позволяющий адаптировать методы построения строгих одномерных функциональных зависимостей для исследования поведения многомерных объектов.

К числу статических моделей, используемых для описания многомерных безынерционных многосвязных объектов относятся многомерные регрессионные зависимости:

$$y_j = \varphi_j(x_{j1}, \dots x_{jk_j}), j = \overline{1; N} \quad (4.10)$$

где  $y_j$  – значение компонента вектора  $Y$  зависимой переменной;

$x_{jl_j}$ ,  $l_j = \overline{1; k_j}$  – компоненты вектора независимых независимой переменной;

$\varphi_j(\cdot)$  – зависимость (параметрическая либо непараметрическая), устанавливающая связь  $j$ -й компоненты вектора зависимой переменной от значений компонент вектора независимой переменной.

Классический подход к построению многомерных регрессионных зависимостей основан на обработке таблицы совместно наблюдаемых значений [94, 107], представленных в табличной форме (см. таблицу 4.1).

Таблица 4.1 – Вид таблицы совместно наблюдаемых значений независимой и зависимой переменных

Номер измерения	Независимые переменные				Зависимая переменная $y_j$
	$x_{j1}$	$x_{j2}$	...	$x_{jl_j}$	
1					
...					
$Q$					

Многомерные уравнения регрессии ассоциируются с многомерными многосвязными объектами, модель которого представлена на рисунке 4.1.

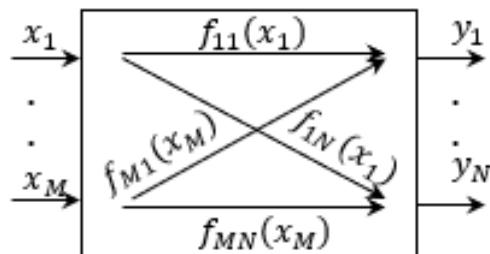


Рисунок 4.1 – Модель многомерного многосвязного объекта

Построение соотношения вида (4.10) в зависимости от формы представления компонент вектора независимой переменной  $x_{jl_j}$  и зависимой переменной  $y_j$  (измерительные данные, либо экспертные оценки) сводится к использованию одного из преобразований (4.4), (4.6), (4.8) для построения одномерных законов распределения случайных величин с последующей оценкой строгих непараметрических функциональных зависимостей на основе (4.2).

Наличие одномерных функциональных зависимостей  $y_j = f_{ij}(x_j)$  позволяет свести формирование таблицы совместно наблюдаемых значений независимой и зависимой переменных к последовательной реализации следующих шагов.

Шаг 1. Генерируются независимые  $q$ -е ( $q = \overline{1; Q}$ ) значения случайных величин  $\xi_j^{(q)}$ , соответствующие оценкам законов распределения  $\hat{F}(x_j)$  на основе датчиков случайных чисел методом обратных преобразований [110]

$$\xi_i^{(q)} = \hat{F}^{-1}(x_j), i = \overline{1; M} \quad (4.11)$$

Шаг 2. По значениям  $\xi_j^{(q)}$ , используя построенные одномерные параметрические зависимости  $f_{ij}(x_j)$  рассчитываются значения  $y_j$  по правилам:

$$y_i^q = \sum_i f_{ij}(\xi_j^{(q)}) \quad (4.12)$$

Шаг 3. Значение  $\{\xi_j^{(q)}, y_i^{(q)}\}$  заносятся в виде строки в таблицу 4.1.

В последующем сформированная таблица обрабатывается известными методами многомерного регрессионного анализа.

### **Пример.**

Допустим имеется двумерный многосвязный объект. Первому входному параметру ставится в соответствие экспертная оценка вида  $x_1 \in [0; 8]$ . Второму входному параметру ставится в соответствие выборка случайных величин, которой соответствует нормальный закон распределения  $N(4,1)$ . Зависимой переменной  $y_1$  ставится в соответствие выборка случайных величин, которой соответствует нормальный закон распределения  $N(8,2)$ . Зависимой переменной  $y_2$  ставится в соответствие экспертная оценка вида  $\{2, [0, \infty)\}$ .

Зависимые и независимые переменные связываются соотношением:

$$\begin{cases} y_1 = f_{11}(x_1) + f_{21}(x_2) \\ y_2 = f_{22}(x_2) + f_{12}(x_1) \end{cases} \quad (4.13)$$

С учетом формы представления экспертной оценки, соответствующей  $x_1$ , в качестве  $F_1(x)$  используем равномерный закон распределения  $F(x_1) = \frac{1}{8} \int_0^{x_1} dt$ .

С учетом, что  $F(y_1)$  описывается нормальным законом распределения, непараметрическая оценка в соответствие с (4.2) приобретает вид, представленный на рисунке 4.2.

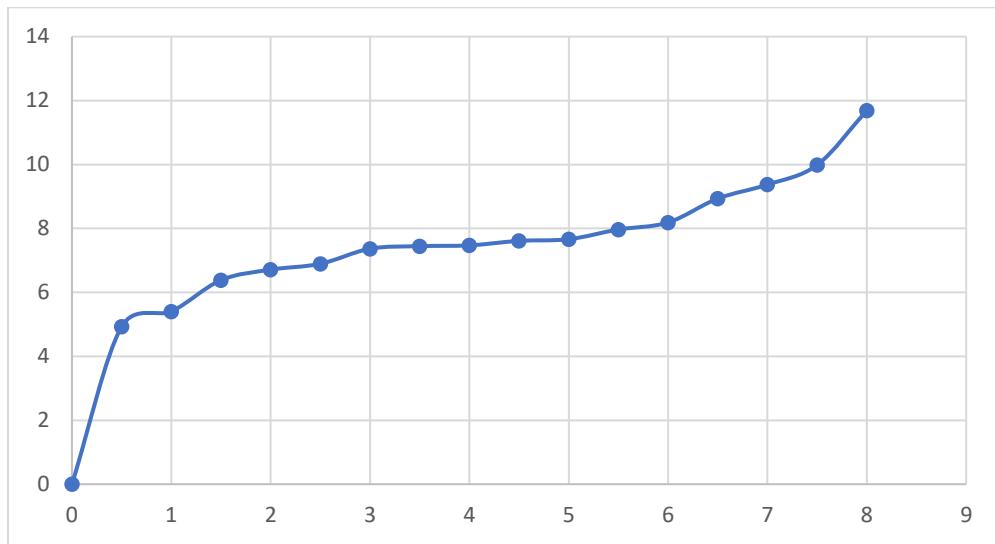
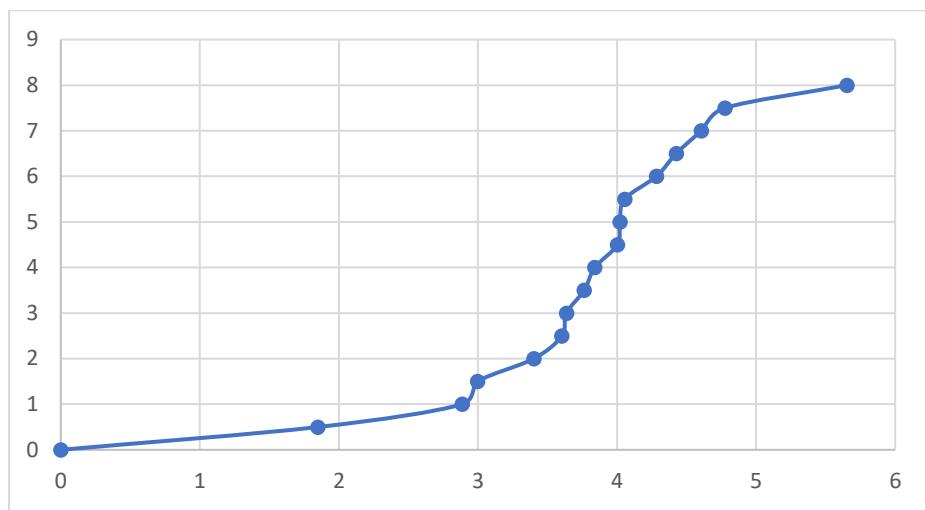
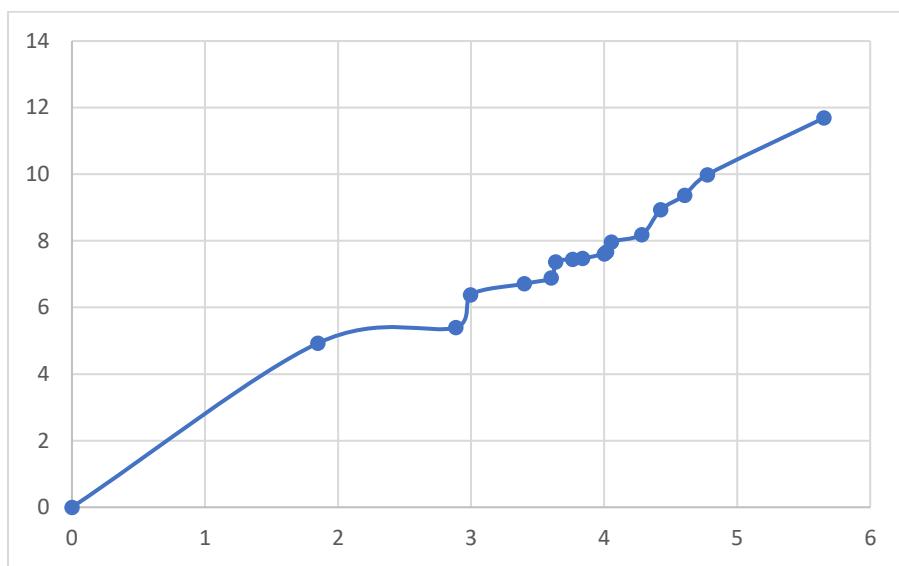
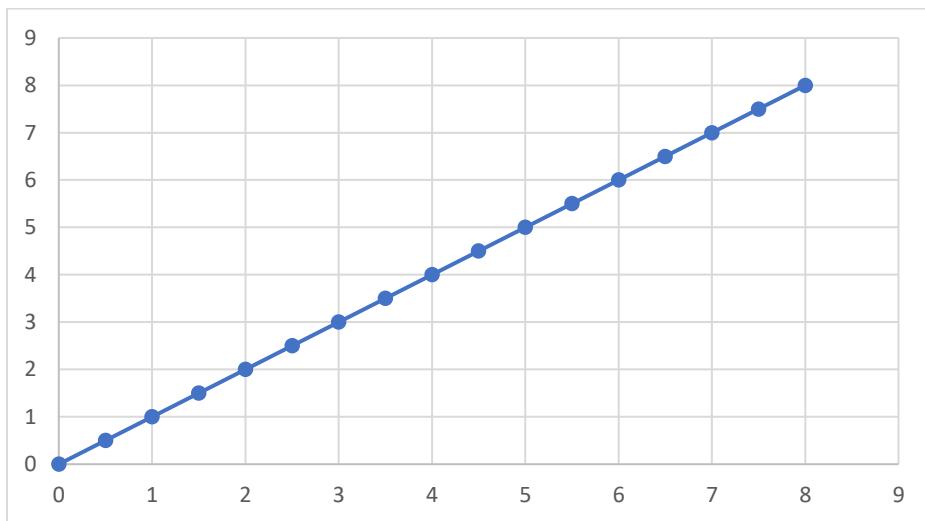


Рисунок 4.2 – Непараметрическая оценка  $f_{11}(x_1)$

С учетом формы представления экспертной оценки, соответствующей  $y_2$ , в соответствии с (4.6) и (4.7) в качестве  $F(y_2)$  используется показательный закон распределения с параметром  $\lambda = 0,5$ .

Учитывая, что закон распределения  $F(x_2)$  является  $N(4,1)$ , непараметрическая оценка  $f_{12}(x_1)$  в соответствие с (4.2) приобретает вид, представленный на рисунке 4.3.

С учётом того, что закон распределения  $F_1(x)$  является равномерным, а  $F(y_2)$  – показательным, использование (4.2) позволяет получить оценку  $f_{12}(x_1)$ , представленную на рисунке 4.4. Учитывая, что  $F(y_2)$  имеют одинаковый тип, использование (4.2) приводит к представлению  $f_{21}(x_2)$  в виде линейной зависимости (рисунок 4.5).

Рисунок 4.3 – Непараметрическая оценка  $f_{22}(x_2)$ Рисунок 4.4 – Непараметрическая оценка  $f_{12}(x_1)$ Рисунок 4.5 – Непараметрическая оценка  $f_{21}(x_2)$

Используя описанный выше трехшаговый алгоритм, сформируем таблицу, содержащую значения независимых и зависимых переменных  $y_1$  и  $y_2$ . При построении таблиц значение  $Q$  принималось  $Q=1000$ .

Фрагмент таблицы совместно наблюдаемых значений независимых и зависимых переменных представлен в таблице 4.2.

Таблица 4.2 – Фрагмент таблицы совместно наблюдаемых значений независимых и зависимых переменных

Номер измерения	$x_1$	$x_2$	$y_1$	$y_2$
1	0	0	0	0
2	0,5	1,849245	4,928309	0,5
3	1	2,88654	5,401828	1
4	1,5	2,996886	6,382905	1,5

На основе таблицы совместно наблюдаемых значений независимых и зависимых переменных построены многомерные функциональные зависимости, представленные на рисунке 4.6 и 4.7

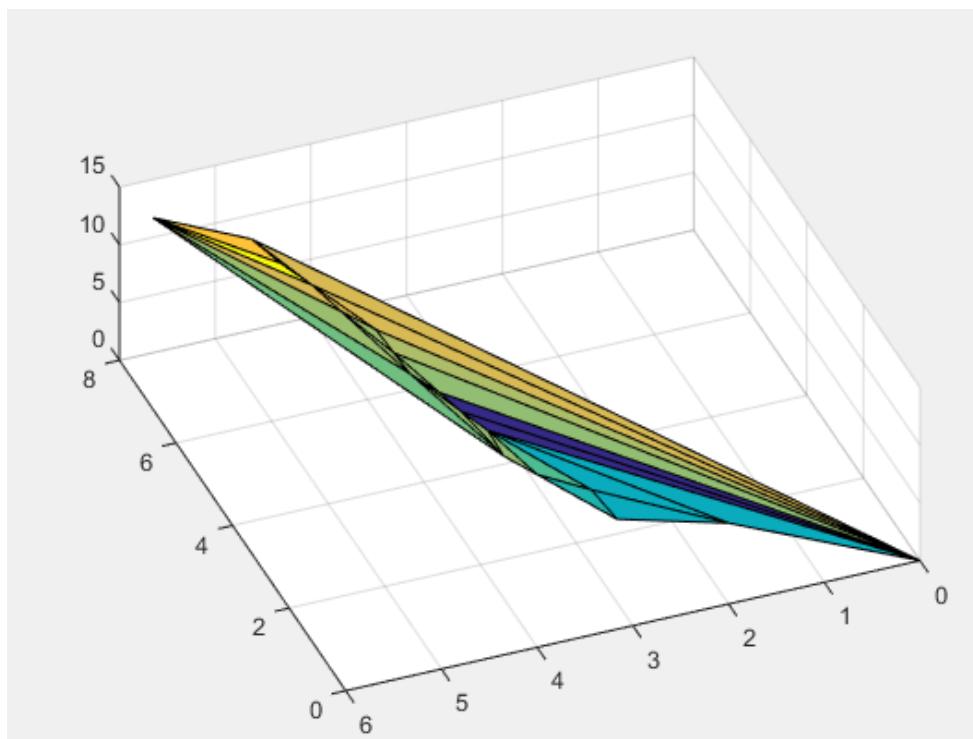


Рисунок 4.6 – Зависимость  $y_1 = \varphi_1(x_1, x_2)$

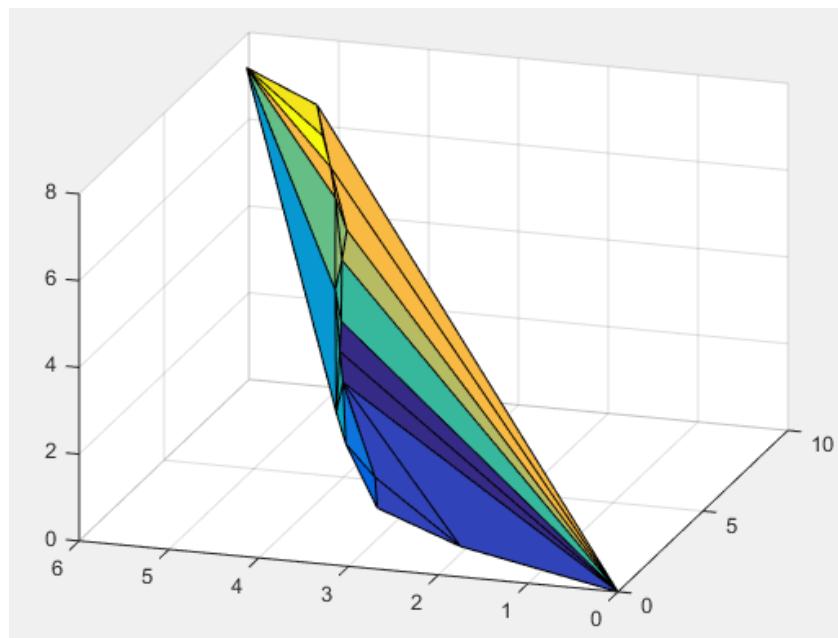


Рисунок 4.7 – Зависимость  $y_2 = \varphi_2(x_1, x_2)$

Таким образом, предложенный метод построения многомерных регрессионных зависимостей, который позволяет представить ИС в виде многосвязного объекта совокупностью одномерных непараметрических регрессионных зависимостей. Основу метода составляет преобразование исходных данных, изначально представляемых в разной форме, к виду закона распределения непрерывной случайной величины. Метод позволяет исследовать поведение многосвязных объектов в случае, когда некоторые компоненты являются метрическими характеристиками, а некоторые представлены экспертными оценками.

Отличительной особенностью предлагаемого подхода является то, что многосвязный объект описывается совокупностью одномерных строгих непараметрических регрессионных зависимостей. Ограничением подхода является необходимость априорного логического обоснования наличия связи между независимыми и зависимыми переменными.

## 4.2 Метод оценки функциональной надежности информационных сервисов на основе концепции профилей с учетом субъективных оценок пользователей

Одним из базовых положений обеспечения функциональной надежности ИС является постоянное отслеживание соответствия потребительских свойств целям и ценностям пользователей, что, в частности, предполагает постоянное отслеживание степени удовлетворенности пользователей надежностью ИС.

Функциональная надежность определяется способностью ИС своевременно предоставлять пользователю полную информацию [116, 207].

При реализации сетецентрического управления и создании единого информационного и управлеченческого пространства целесообразно рассматривать не функции, а информационные сервисы.

Сервис-ориентированная архитектура (*SOA*) – это подход к разработке программного обеспечения, при котором система строится как набор слабо связанных сервисов [114]. Эти сервисы взаимодействуют друг с другом через чётко определённые интерфейсы.

Основные принципы *SOA*:

- Независимость сервисов. Сервисы должны быть независимы друг от друга и разрабатываться отдельно. Это позволяет изменять или заменять один сервис, не затрагивая другие.
- Слабая связанность. Связь между сервисами должна быть минимальной. Это обеспечивает гибкость системы и возможность её масштабирования.
- Абстракция. Сервисы предоставляют абстрактные операции, которые могут быть использованы другими сервисами. Это упрощает взаимодействие между ними.
- Стандартизация. Для взаимодействия сервисов используются стандартные протоколы и форматы данных. Это делает систему открытой и совместимой с другими системами.

Сервис-ориентированная архитектура позволяет быстро разрабатывать новые приложения, интегрировать существующие системы и обеспечивать их гибкость и масштабируемость, является основой гибкого переконфигурирования сети при изменении бизнес-процессов.

В литературе описан подход к анализу надежности программных систем, концептуальную основу которого составляет стремление «... взглянуть на надежность программных систем глазами пользователя» [86].

Объектом исследования является функциональная надежность информационных сервисов. Основная идея подхода заключается в учете восприятия свойств сервисов пользователями для того, чтобы определить, как эти сервисы улучшить.

Особенность такого подхода заключается в учете субъективной составляющей при оценивании надежности одних и тех же сервисов разными пользователями. Субъективизм обуславливается, во-первых, различным восприятием результатов сервисов различными субъектами, а во-вторых, тем, что рассматриваемый сервис может быть в разной степени полезен при решении разных бизнес-задач.

Одним из факторов, который должен учитываться при построении профиля информационных сервисов, является учет множества режимов функционирования ИС (штатного, предпосылки возникновения нештатного режима; нештатного). Кроме того, в разных режимах возникают разные информационные потребности потребителей сервисов, что обуславливает изменение ими оценок свойств одних и тех же сервисов в разных режимах.

Под профилем информационного сервиса (пример показан на рисунке 4.8) понимается полное множество альтернатив (например, множество альтернативных категорий пользователей, функций и т.п.), для каждой из которых существует вероятность появления.

Формально профиль представляет собой дерево, каждому ребру которого ставится в соответствие весовая характеристика, характеризующая вероятность перехода по нему.

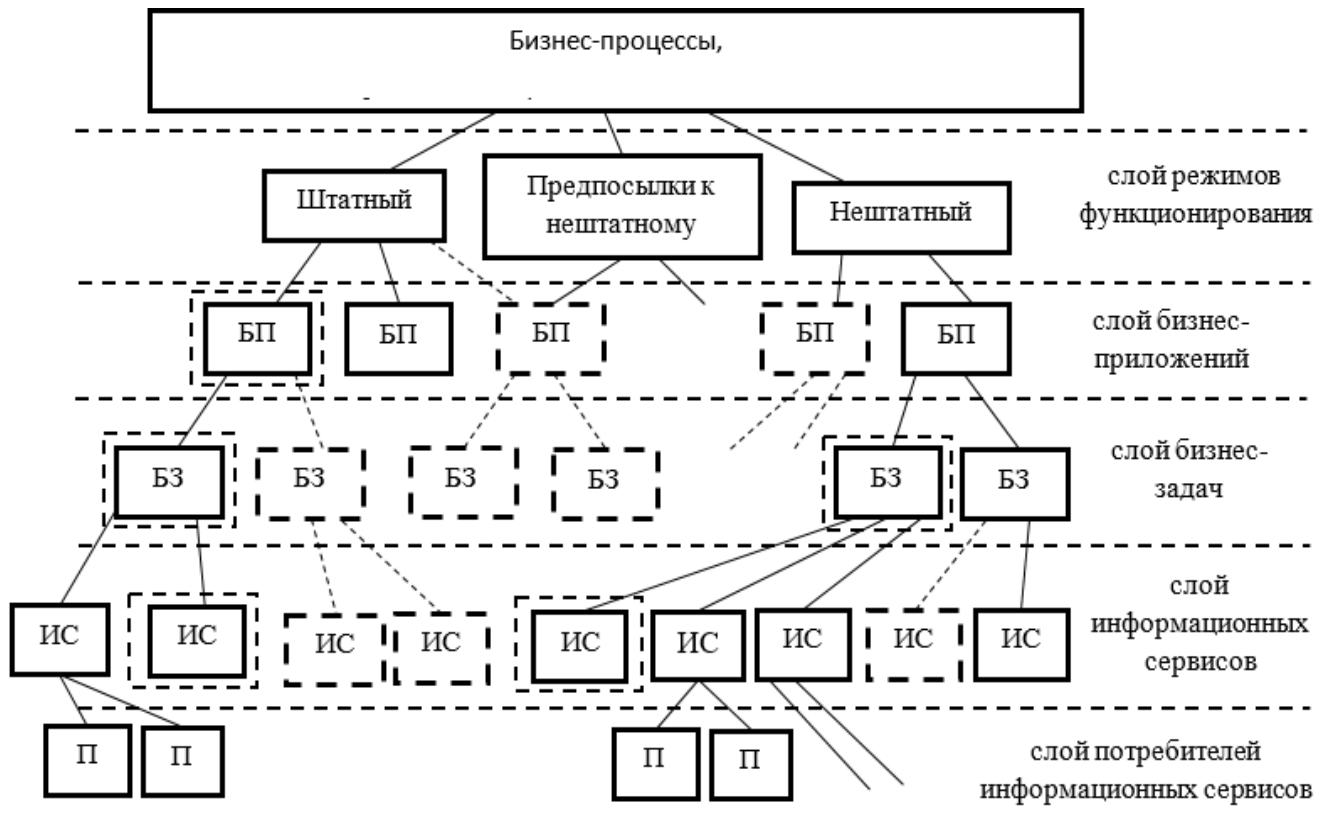


Рисунок 4.8 – Профиль информационных сервисов

Так как предполагается, что переход по одному и только по одному из ребер из каждой вершины происходит обязательно, а по какому именно – заранее неизвестно, каждому из узлов ставится в соответствие соотношение из формулы (4.14).

$$\sum_j P_{i,j}^{(l)} = 1 \quad (4.14)$$

где  $l$  – номер слоя;

$i$  - номер узла в слое;

$j$  – номер ребра, исходящего из  $i$ -го узла.

Построение профиля сервисов позволяет дать количественную оценку степени покрытия бизнес-задач информационными сервисами. Так, для  $i$ -ой задачи слоя бизнес-задач степень покрытия определяется по формуле:

$$CPI_i^{(BZ)} = \frac{IC_i^{(P)}}{IC_i^{(P)} + IC_i^{(PP)}}, \quad (4.15)$$

где  $IC_i^{(P)}$  – число реализованных информационных сервисов;

$IC_i^{(PP)}$  – число информационных сервисов, предполагаемых к реализации.

Аналогичным образом можно провести оценивание для всех режимов работы. Одной из важнейших характеристик, которую необходимо учитывать при распределении ресурсов, в частности временных, на разработку новых информационных сервисов и улучшение уже существующих, является степень покрытия бизнес-задач информационными сервисами.

Различным вариантам профилей сервисов можно поставить в соответствие системную модель вида:

$$\langle E, Pr, Rel \rangle, \quad (4.16)$$

где  $E$  – множество возможных событий (простых путей в графе);

$Pr$  – множество значений вероятности (неявных профилей), соответствующих каждому из событий (неявный профиль определяется как произведение весов дуг, входящих в простой путь);

$Rel$  – множество оценок надежности информационных сервисов.

При построении формальных моделей возникает потребность в преобразовании субъективных оценок пользователей к виду количественных оценок.

Для оценивания характеристик надежности на основе субъективных оценок пользователей предлагается использовать подход, основанный на функции принадлежности лингвистической шкалы.

Лингвистическая шкала в терминах, которой пользователи, принадлежащие к одной и той же  $m$ -й целевой группе, оценивают надежность информационного сервиса имеет вид:  $\{\text{низкая; средняя; высокая}\}$ .

Каждому  $l$ -му значению лингвистической шкалы ставится в соответствие функция принадлежности  $\mu_l$ , определенная на интервале  $\mu_l \in [0;1]$ .

Положение максимума принадлежности  $l$ -го значения лингвистической шкалы определяется как его опорное значение  $r_l$  на оси  $y$  (рисунок 4.9).

Каждый потребитель выражает свою точку зрения о надежности информационного сервиса и своей степени уверенности в этом.

Постулируется, что оценки, даваемые пользователями, отражают его истинное независимое мнение относительно информационного сервиса.

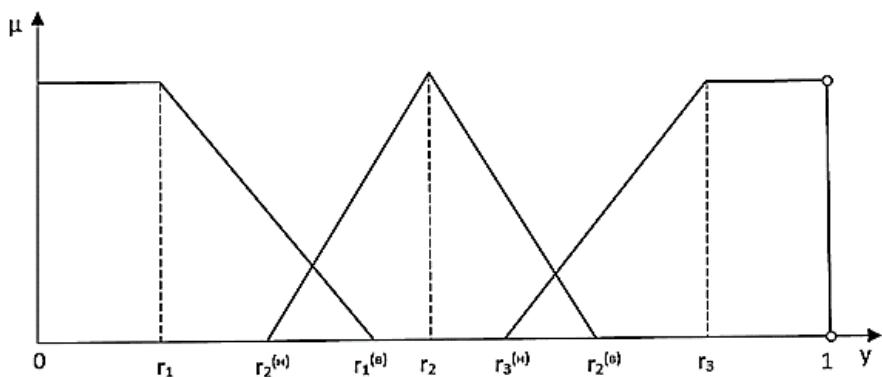


Рисунок 4.9 – Качественный вид функции принадлежности лингвистической шкалы

Основной вопрос задачи квантификации экспертных оценок: выбор функций принадлежности. Предлагаемое решение: функции принадлежности треугольного вида с одинаковыми границами, совпадающими с границами оси  $y \in [0;1]$ .

Совокупная оценка надежности на основе субъективных оценок пользователей определяется на основе соотношения:

$$D^{(m)} = \frac{\sum_{k=1}^{N_m} \mu_k^{(l)} * r_l^{(k)}}{\sum_{k=1}^{N_m} \mu_k^{(l)}}, \quad (4.17)$$

где  $N_m$  – количество пользователей, принадлежащих  $m$ -й целевой группе;

$r_l^{(k)}$  – опорное значение, соответствующее  $l$ -му значению лингвистической шкалы, определенному  $k$ -м пользователем;

$\mu_k^{(l)}$  – степень уверенности  $k$ -го пользователя в даваемой оценке.

На рисунке  $\{r_l^{(n)}, r_l^{(b)}\}$  – нижняя и верхняя границы, в которых на оси  $y$  определена  $l$ -я функция принадлежности.

Основными неопределенностями затрудняющие практическое использование (4.17) являются:

- подходы к выбору значений  $r_l$ ;
- подходы к выбору формы функции принадлежности  $\mu_l$ . В литературе, например, в работе [60], приведено описание большого числа функций принадлежности разной формы, однако не определены четкие практические рекомендации по их применению;

– подходы к назначению границ  $\{r_l^{(n)}, r_l^{(v)}\}$  (см. рис. 4.9).

Для устранения выделенных неопределенностей предлагается использовать следующее.

В основу определения  $r_l$  положить учет доли пользователей, принадлежащих  $m$ -й целевой группе и выбравших  $l$ -е значение лингвистической шкалы.

При таком подходе в качестве  $r_l$  выступает величина  $\frac{n_1^{(m)}}{N_m}$ ,

где  $n_1^{(m)}$  – число пользователей  $m$ -й целевой группы, выбравших значение лингвистической шкалы «низкая».

В качестве  $r_2$  выступает величина:  $r_2 = \frac{n_1^{(m)}}{N_m} + \frac{n_2^{(m)}}{2N_m}$ ,

где  $n_2^{(m)}$  – число пользователей целевой группы, выбравших значение лингвистической шкалы «средняя».

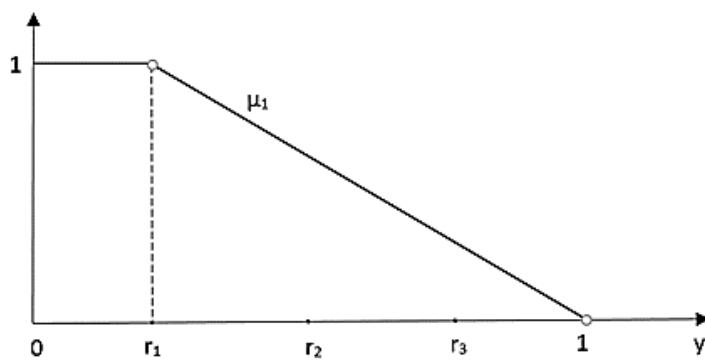
Значение  $r_3$  определяется соотношением:  $r_3 = \frac{n_1^{(m)}}{N_m} + \frac{n_2^{(m)}}{N_m}$ . Величина  $r_2$

представляет собой середину подинтервала между значениями  $r_1$  и  $r_3$ .

Относительно неопределенностей, связанных с выбором формы функции принадлежности, а также нижней и верхней границ  $\{r_l^{(n)}, r_l^{(v)}\}$ , следует отметить следующее.

Если значению  $r_l$  соответствует максимальное значение лингвистической шкалы «низкая», то левее  $r_l$  степень удовлетворенности увеличиться не может, т.е.  $\mu_1$  при  $y \in [0; r_l]$  принимает значение «единица». Правее точки  $r_l$  уверенность эксперта в выбранной оценке падает. В силу того, что какие-либо сведения о скорости уменьшение значения  $\mu_1$  по мере удаления вправо от точки  $r_l$  отсутствуют, следуя принципу максимизации энтропии [59] целесообразно считать эту скорость постоянной.

В силу отсутствия сведений, накладывающих ограничения на область определения  $\mu_1$ , в качестве такой области целесообразно принять весь интервал  $y \in [r_l; 1]$ . С учетом высказанных соображений  $\mu_1$  принимает вид, представленный на рисунке 4.10.

Рисунок 4.10 – Вид функции принадлежности  $\mu_1$ 

Рассуждая по аналогии, функции принадлежности для выбранной лингвистической шкалы имеют вид, представленный на рисунке 4.11.

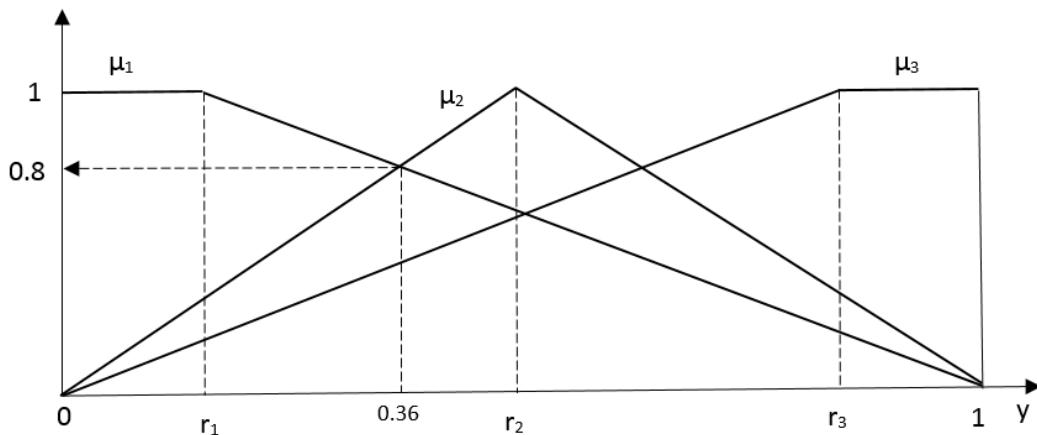


Рисунок 4.11 – Итоговая форма функции принадлежности лингвистической шкалы

*Рассмотрим пример использования предлагаемого подхода для квантификации экспертных оценок.*

Допустим, в результате опроса десяти пользователей, принадлежащих одной и той же целевой группе, получены следующие оценки надежности информационных сервисов и степень уверенности в даваемых оценках:

{низкая; 0,8}, {средняя; 0,95}, {средняя; 0,7}, {высокая; 0,4}, {средняя; 0,8}, {низкая; 0,7}, {средняя; 0,8}, {средняя; 0,6}, {высокая; 0,7}, {высокая; 0,9}.

В соответствие с (4.17) значение  $D$  составит:

$$D^{(m)} = \frac{0,2(0,8+0,7)+0,45(0,95+0,7+0,8+0,8+0,6)+0,3(0,4+0,7+0,9)}{0,8+0,7+0,95+0,7+0,8+0,85+0,6+0,4+0,7+0,9} = 0,358$$

Ближайшим опорным значением к  $D^{(m)}$  оказывается  $r_2=0,45$ .

Значению  $D^{(m)}$  соответствует значение  $\mu_m(0,36) = 0,8$  (см. рис.4.11).

Таким образом, разработанный подход позволяет получать метрические оценки надежности информационных сервисов на основе совокупного использования как измерительных данных, так и субъективных оценок пользователей.

Практической направленностью предлагаемого метода является определение приоритетных направлений совершенствования функциональной надежности. Следует обратить внимание, что предлагаемый подход к квантификации экспертных оценок является в значительной степени формализованным, что позволяет реализовать его в качестве инструментального средства.

#### **4.3 Метод структурного моделирования последствий отказов функциональных компонент информационных систем**

На предпроектной стадии обеспечение функциональной надежности ИС предполагает анализ альтернатив. Поскольку на начальных стадиях выполняется только построение структурных схем, поэтому сравнительный анализ альтернативных решений по критерию функциональной надежности сводится к сопоставлению возможных последствий отказов (что является реализацией сценарного подхода).

В данном разделе описывается теоретическая основа подхода, основанного на адаптации аппарата схем сопряжения к задачам обеспечения функциональной надежности ИС на предпроектной стадии.

В работе [84] показано, что потребительские свойства системы определяются ее устройством, а также свойствами окружающей ее среды. Устройство системы можно описать посредством описания свойств компонентов системы и связей между ними [186].

На предпроектной стадии в силу необходимости обеспечения рассмотрения физических и программных компонентов ИС на одном уровне абстракции, целесообразно компоненты представлять в виде «черного ящика».

Связь входных и выходных параметров компонентов целесообразно описывать посредством таблиц истинности.

Для унификации описания связей между физическими и программными компонентами ИС целесообразно использовать некое абстрактное понятие, инвариантное к физической природе связей. Известен подход к описанию структуры системы посредством аппарата схем сопряжения агрегатов [31].

Агрегат представляет собой математическую схему общего вида. В качестве частных случаев можно привести функции Булевой алгебры, системы массового обслуживания, модели конечных автоматов, различные динамические системы, которые можно описать при помощи дифференциальных уравнений и др.

Таким образом, агрегат является универсальной моделью объекта, служащего для приема входной и управляющей информации, ее обработки и выдачи выходной информации.

В общем случае система рассматривается как многоуровневая конструкция из взаимодействующих элементов, при необходимости объединяемых в разные подсистемы. Взаимодействие элементов в процессе функционирования системы рассматривается как результат совокупности воздействий между элементами посредством сигналов.

Аппарат позволяет в единой нотации описывать системы на разных уровнях абстракции: отдельных функций; подсистем; системы. Взаимодействие системы с внешней средой сводится к механизму обмена сигналами между элементами системы и объектами внешней среды, совершенно аналогичному тому, который описывает взаимодействие элементов внутри системы.

В качестве базы для сопоставления свойств физических устройств и программных компонентов по критерию функциональной надежности выступает понятие «отказ». В данном случае под отказом понимается проявление латентных, то есть скрытых, дефектов.

Конкретное содержание этого понятия не приводится по той причине, что оно должно определяться назначением функциональных компонентов ИС. Такое толкование понятия «отказ» не противоречит положениям ГОСТ 24.104-85 [49].

В качестве технологической основы для построения различных статических структурных моделей отказов и их последствий предлагается совместно использовать схемы сопряжения и таблицы истинности.

Совместное использование схем сопряжения и таблиц истинности создает основу для построения структурных моделей, в рамках как нисходящего (дерево неисправностей *FTA*, структурная модель выявления коренных причин отказов на основе подхода *RCA*, модели причинно-следственных связей *Fishbone*), так и восходящего подходов к исследованию функциональной надежности (дерево событий *ETA*). Также данный подход создает основу для исследования (в рамках реализации концепции профилей) последствий нарушения функционирования разных компонентов, входящих в состав системы, на функционирование других компонентов.

Метод структурного моделирования последствий отказов включает в себя следующую последовательность:

- 1) построение схемы сопряжения агрегатов;
- 2) построение таблиц истинности;
- 3) построение на основе схемы сопряжения структурной модели отказов в рамках нисходящего подхода;
- 4) построение на основе схемы сопряжения структурной модели отказов в рамках восходящего подхода;
- 5) выявление коренных причин отказов методом *RCA*;
- 6) построение структурных моделей в рамках концепции профилей.

### *Пример реализации*

Пусть известен возможный сценарий  $S_0$  функционирования ИС.

Этот сценарий представляет собой упорядоченную во времени последовательность реализации логически завершенных этапов. Каждый из этапов представляет собой совокупность взаимосвязанных и взаимодействующих функциональных компонентов.

Схема сопряжения функциональных компонентов ИС представлена на рисунке 4.12. На рисунке 4.12 обозначены:

$x_{km}$  – входные контакты компонентов и внешней среды;

$y_{km}$  – выходные контакты компонентов и внешней среды;

$k$  – признак компонента (для внешней среды  $k=0$ );

$m$  – признак входа/выхода компонента.

В свою очередь, каждому из компонентов ставится в соответствие таблица истинности (примеры таблиц истинности, соответствующих компонентам  $K1$  и  $K8$ , представлены на рисунке 4.13).

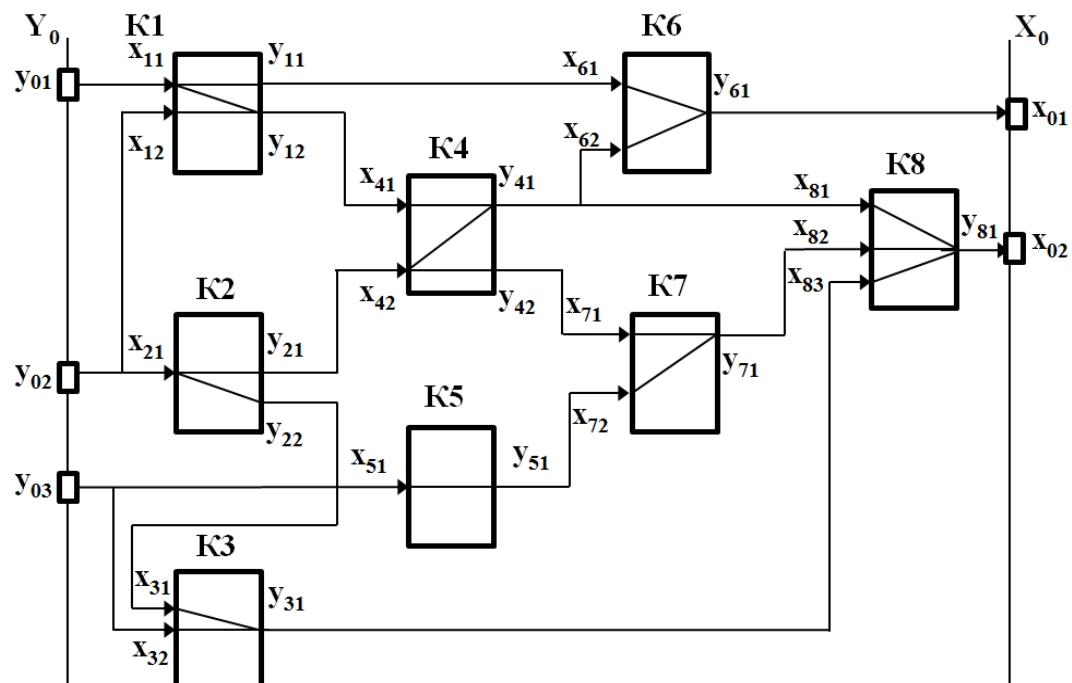


Рисунок 4.12 – Схема сопряжения функциональных компонентов ИС

В таблицах «0» соответствует ошибке, «1» корректному состоянию.

K1	inputs		outputs		K8	inputs		outputs	
	X <sub>11</sub>	X <sub>12</sub>	Y <sub>11</sub>	Y <sub>12</sub>		X <sub>81</sub>	X <sub>82</sub>	X <sub>83</sub>	Y <sub>81</sub>
0	0	0	0	0		0	0	0	0
0	1	0	0	0		0	0	1	0
1	0	1	1	0		0	1	0	0
1	1	1	1	1		0	1	1	0

Рисунок 4.13 – Таблицы истинности

На рисунке 4.14 представлена структурная модель, составляющая основу моделирования отказов в рамках нисходящего подхода *FTA* [196].

На рисунке 4.15 представлена структурная модель, составляющая основу моделирования отказов в рамках восходящего подхода *ETA*.

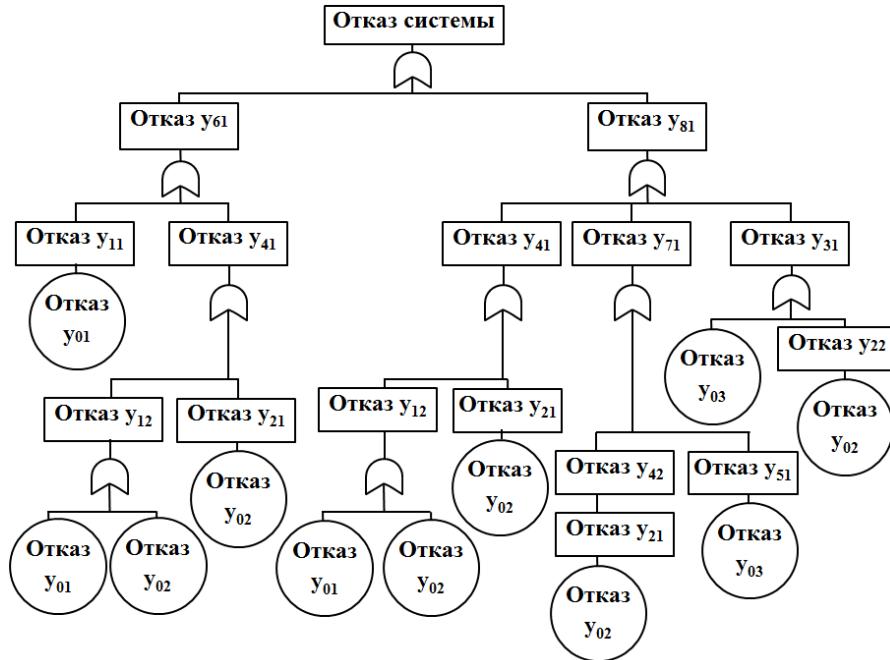


Рисунок 4.14 – Дерево отказов, соответствующее схеме сопряжения (*FTA*)

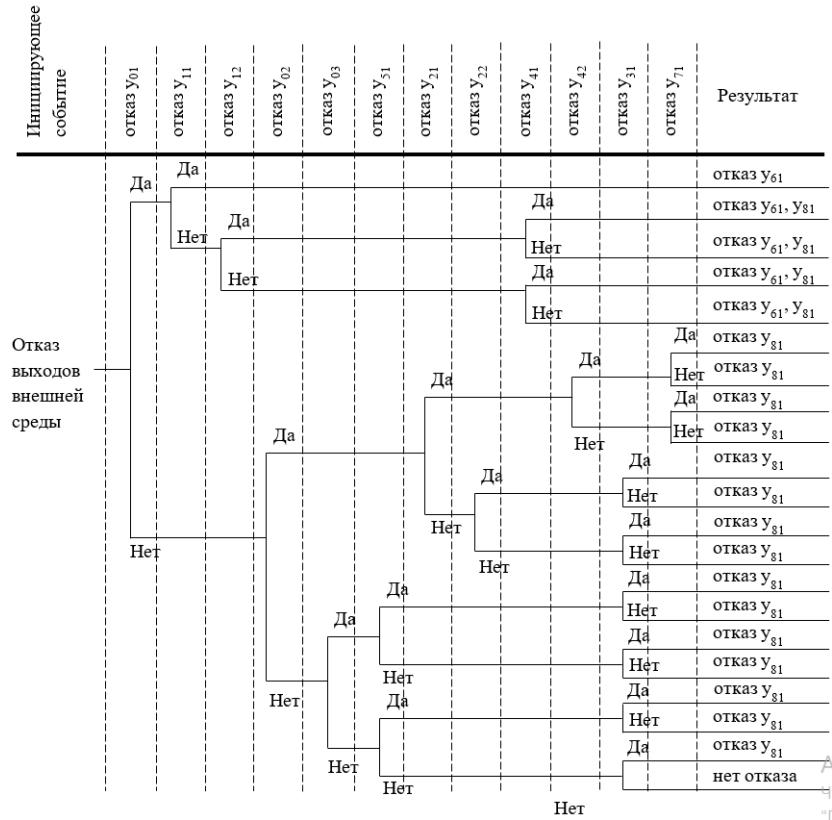


Рисунок 4.15 – Дерево событий, соответствующее схеме сопряжения (*ETA*)

На рисунке 4.16 представлена структурная модель, составляющая основу выявления возможных коренных причин отказов на основе подхода *RCA*.

Выделенные слои имеют следующий смысл:

слой 1 – слой симптомов;

слой 2 – слой непосредственных причин;

слой 3 – слой причин первого порядка;

слой 4 – слой причин второго порядка;

слой 5 – слой причин третьего порядка.

В рамках этой модели отражено то обстоятельство, что симптомы отказов, наблюдаемые на выходах этапа ( $y_{61}$  и/или  $y_{81}$ ), могут быть обусловлены не только неверными значениями сигналов на входах компонентов системы, но и нарушением внутренней логики компонентов  $VLK_i$  ( $i = \overline{1;8}$ ) из-за активизации латентных дефектов, имеющихся в компонентах.

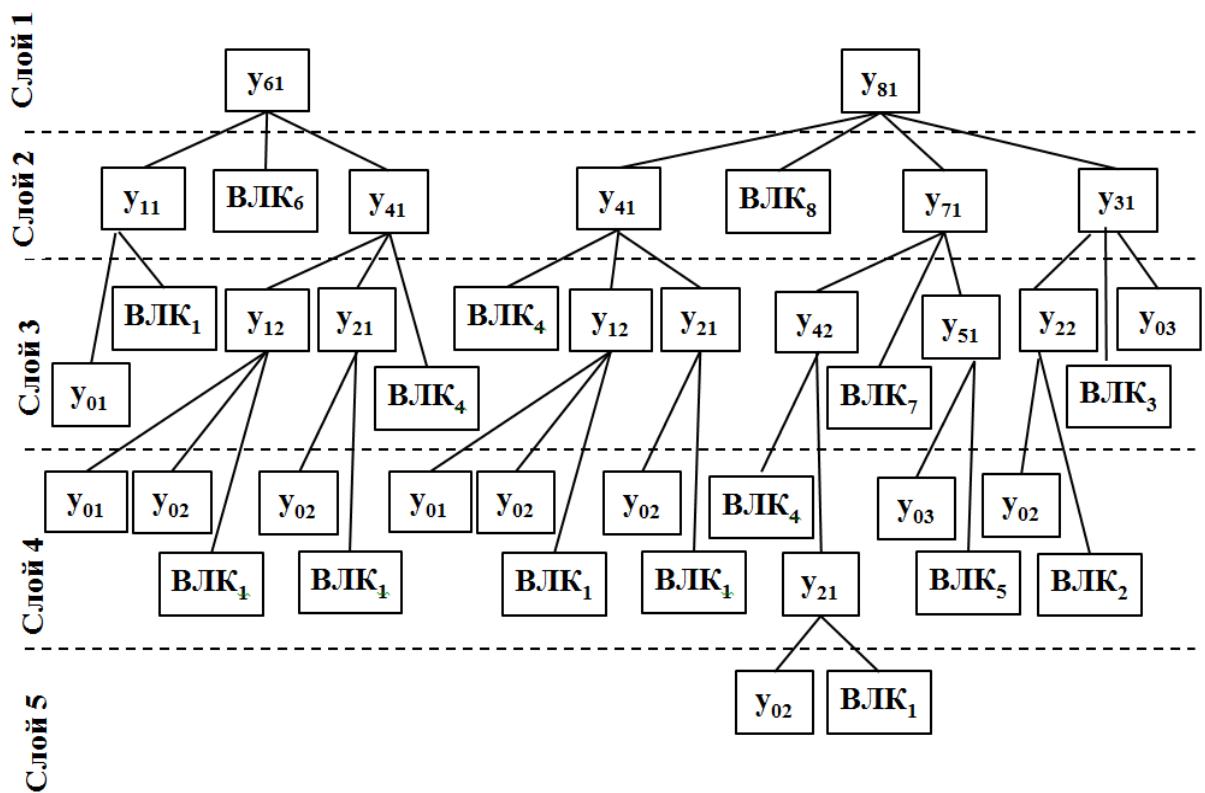


Рисунок 4.16 – Выявление коренных причин отказов методом *RCA*

На рисунке 4.17 представлены *FishBone* – диаграммы, являющиеся иной формой представления модели, представленной на рисунке 4.16.

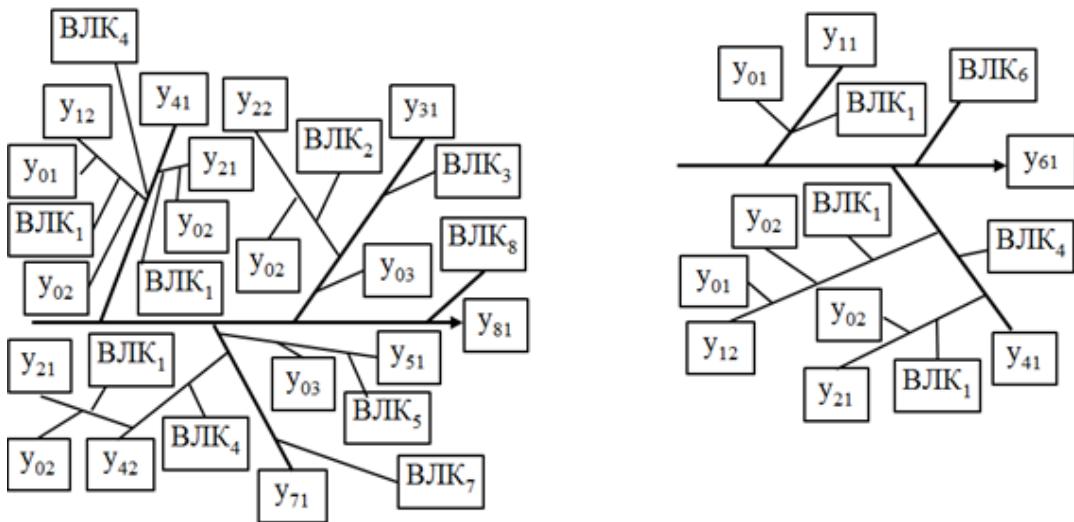


Рисунок 4.17 – Модели причинно-следственных связей, соответствующие модели анализа коренных причин

На рисунке 4.18 представлена модель, построенная на основе схемы сопряжения в рамках концепции профилей.

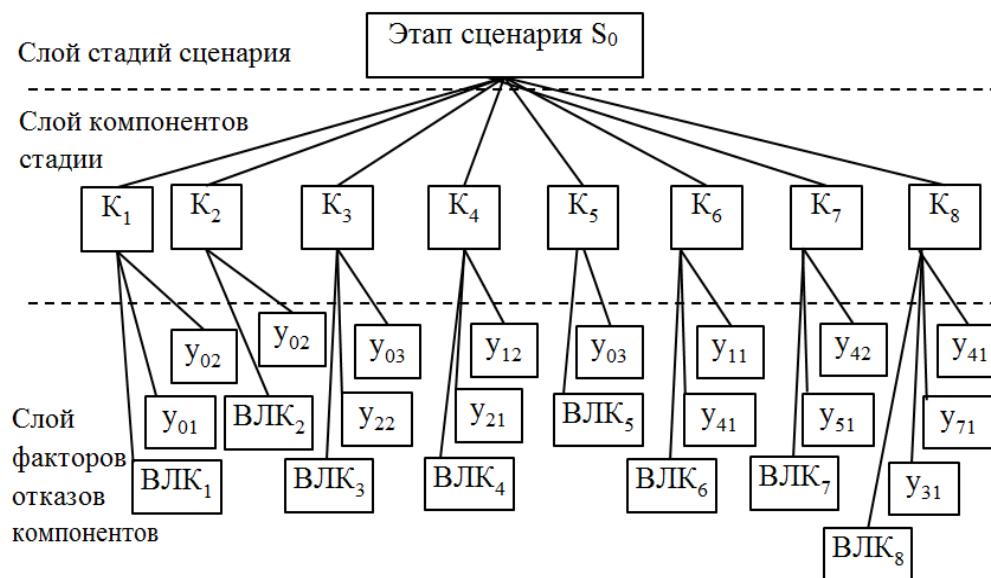


Рисунок 4.18 – Профиль непосредственных причин отказов компонентов

Профиль, представленный на рисунке 4.19, характеризует количество факторов, определяющих надежное функционирование каждого из компонентов системы, представленной на схеме сопряжения (рисунок 4.12).

Профиль, представленный на рисунке 4.20, характеризует число компонентов, на которые передаются сигналы с  $l$ -го выхода  $k$ -ой компоненты ( $l, k$ ).

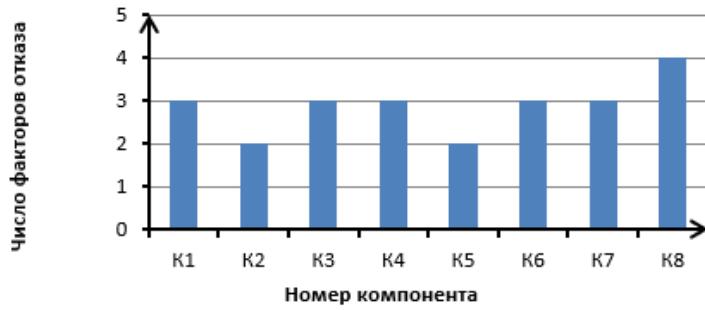


Рисунок 4.19 – Профиль числа факторов, определяющих надежное функционирование каждого из компонентов



Рисунок 4.20 – Профиль числа компонентов, которые оказываются под влиянием данного фактора

На рисунке 4.21 представлена модель, позволяющая определить транзитивное влияние каждого из компонентов системы  $K_i$  ( $i = \overline{1; 8}$ ) на выходы других компонентов.

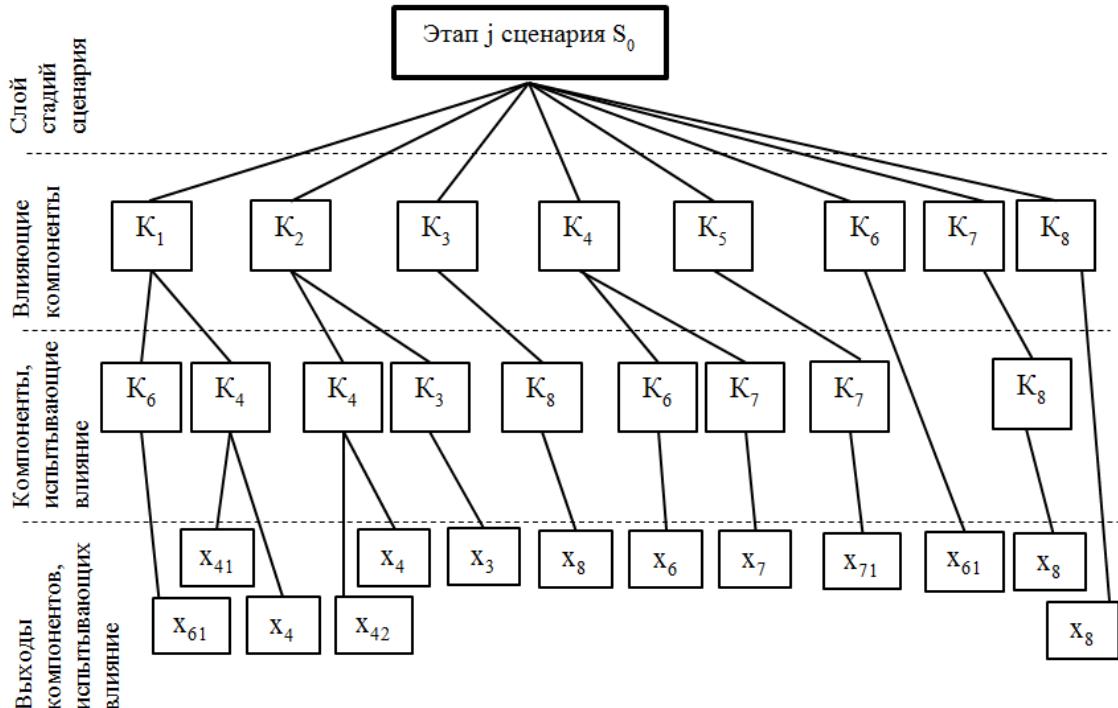


Рисунок 4.21 – Профиль транзитивного влияния компонентов

Построение профилей создает основу для принятия управленческих решений по распределению ресурсов проекта с учетом влияния факторов на состояние компонентов, включая функциональную надежность.

Рассмотренный пример позволяет заключить, что описание систем посредством совместного использования схем сопряжения и таблиц истинности создают базу для разноспектрного анализа функциональной надежности ИС на предпроектной стадии.

Предложенный метод создает информационную основу для сравнения альтернативных вариантов проектных решений по критерию функциональной надежности ИС на предпроектной стадии.

#### **4.4 Диверсионный анализ. Схема анализа событий в рамках AFD-1**

Разноспектрное исследование проблемы обеспечения функциональной надежности ИС на предпроектной стадии предполагает использование различных подходов и методов.

Метод, известный в зарубежной литературе под названием *Anticipatory Failure Determination (AFD)* [178, 208, 212, 213], а в отечественной практике – как диверсионный анализ, представляет собой один из перспективных подходов к формированию систематизированной процедуры разработки защитных барьеров. Применение подхода *AFD* к ИС направлено на идентификацию дефектов различной природы, способных трансформировать потенциальные опасности в нежелательные события [178, 191, 204].

Содержанием *AFD* охватывает решение задач трех классов:

- 1) Выявление потенциальных опасностей, включая проектные и организационные ошибки, опасные действия персонала, отказы оборудования, климатические воздействия и иные негативные факторы.
- 2) Выявление возможных каузальных цепочек, описывающих причинно-следственные связи событий, связанных с преобразованием опасностей в инциденты, а также выделение условий, при которых возможна их реализация.

3) Разработка реалистичных и эффективных мер (барьеров), препятствующих реализации каузальных цепочек.

Основополагающие принципы *AFD* и их содержание приедставлены в таблице 4.3.

Таблица 4.3 - Основополагающие принципы *AFD*

№	Принцип	Оригинальное название (англ.)	Содержание принципа
1	Принцип успешного сценария	<i>The Principle of S<sub>0</sub></i>	Основа сценариев инцидентов – четкое различие желаемой траектории состояния системы ( $S_0$ ) и альтернативных траекторий, возникающих из внешних и внутренних событий.
2	Принцип наличия инициирующего события	<i>The Principle of Initiation</i>	Отклонение сложной системы от желаемой траектории не возникает без причины.
3	Принцип множественности последствий инициирующих событий	<i>The Principle of Emanation</i>	Один и тот же инициирующий фактор может приводить к различным, в том числе позитивным и негативным, исходам.
4	Принцип бесконечности формирования причинно-следственных связей	<i>The Principle of Unending Cause-Effect</i>	Понятия «инициирующее событие» и «конечное состояние» условны: инициирующее событие может быть результатом более ранней цепочки, а конечное состояние – началом новой на другом уровне абстракции.
5	Принцип детализации	<i>The Principle of Subdivision</i>	Любой компонент сценария может быть представлен как сценарий более низкого уровня.
6	Принцип эквивалентных траекторий	<i>Pinch Point Principle</i>	К одному и тому же состоянию системы могут вести разные каузальные цепочки.
7	Принцип равной важности знаний о причинах и последствиях событий	<i>Fault and Event Trees</i>	Для предотвращения нежелательных событий одинаково важны анализ происхождения цепочки и оценка возможных последствий предполагаемого инициирующего события; требуются и эмпирические знания разработчиков барьеров, и способность предвидеть новые инциденты.
8	Принцип достаточности ресурсов	<i>The Principle of Resources</i>	Событие реализуется лишь при одновременном наличии всех необходимых ресурсов в нужном месте и времени; отсутствие хотя бы одного ресурса блокирует развитие цепочки. Ресурсы могут синтезироваться из вторичных составляющих системы или окружающей среды.

В рамках *AFD* различают два подхода. *AFD-1* ориентирован на анализ нежелательных событий, которые уже имели место. *AFD-2* ориентирован на прогнозирование еще не реализовавшихся, но возможных событий.

Концептуально *AFD-1* сдвигает фокус с вопроса «Почему произошло нежелательное событие?» на «Каким способом это могло быть реализовано?», тогда как *AFD-2* смещает акценты с вопроса «Что может пойти не так?» на «Каким образом эффективнее всего нанести ущерб заданного масштаба?». В этом заключается суть диверсионного анализа, сочетающего реактивные и проактивные подходы к обеспечению функциональной надёжности систем.

Из содержания основополагающих принципов *AFD* можно сделать вывод о сходстве этого подхода с другими известными в рамках системного анализа подходами. Так, например: содержание принципа успешного сценария по смыслу коррелирует с базовым положением теории надежности, согласно которому основу исследования свойств технических систем по критериям надежности составляет формальное определение понятия «отказ», а содержание принципа детализации представляет собой нисходящий подход к исследованию систем;

Таким образом, системный анализ представляет собой методологическую основу *AFD*. Это обосновывает возможность адаптации моделей и методов системного анализа к задачам качественного и количественного исследования ИС в рамках *AFD*.

Схема анализа событий в рамках *AFD-1* в общем случае сводится к реализации следующей последовательности шагов:

Шаг 1. Формулировка проблемы. Назначение этого шага – характеристика исследуемой системы, описание симптомов и непосредственных причин («спусковых механизмов») событий.

Шаг 2. Описание идеального сценария  $S_0$ .

На данном шаге выделяются логически завершенные стадии процесса с описанием результатов, которые должны получаться в конце каждой стадии.

Шаг 3. Реализация области события.

Назначением этого шага является определение фазы процесса/компоненты системы, в которой/котором произошло событие. На этом шаге сокращается область поиска, т.е. исключается из рассмотрения функции/компоненты системы, в которых не могло произойти событие.

Шаг 4. Формулирование расширенной инвертированной проблемы в общем виде. На этом шаге решаются следующие задачи:

4.1. Выявляются непосредственные причины и условия, приведшие к наблюдавшимся событиям.

4.2. Выявляются способы и условия реализации событий равных и превосходящих по последствиям те, что наблюдались.

Шаг 5. Поиск способов реализации инвертированной проблемы.

На этом шаге осуществляется поиск того, как можно создать причины и условия, увеличивающие возможность проявления нежелательных событий. Реализация этого шага предполагает получение ответов на следующие вопросы:

5.1. Поиск очевидных решений. Существо этого вопроса сводится к определению функций/компонентов системы, события в которых будут решениями инвертированной проблемы.

5.2. Идентификация ресурсов. Определяется перечень ресурсов, необходимых для реализации очевидных решений.

5.3. Поиск способов получения ресурсов.

Шаг 6. Формулирование гипотезы по способам реализации инвертированной проблемы.

На этом шаге формулируются гипотезы о способах решения задач, определенных на шаге 5.

Шаг 7. Определение действий, препятствующих реализации инвертированной проблемы.

На этом шаге определяются барьеры, препятствующие реализации гипотез, определенных на шаге 6.

### ***Пример***

В составе системы управления имеется программный продукт, реализующий вычисление ряда функций (рисунок 4.22).

Исходные данные (значения  $x$ ), поступающие от объекта управления, являются вещественными величинами. По оценкам экспертов область фактических значений  $x$ , соответствующих штатному режиму функционирования объекта управления, являлись  $x \in (3.5; 4.5)$ .

В техническом задании на разработку программного продукта область возможных значений определена как  $x \in (3; 5)$ , чему соответствуют значения  $y_4 \in (y_4^{(h)}, y_4^{(e)})$ .

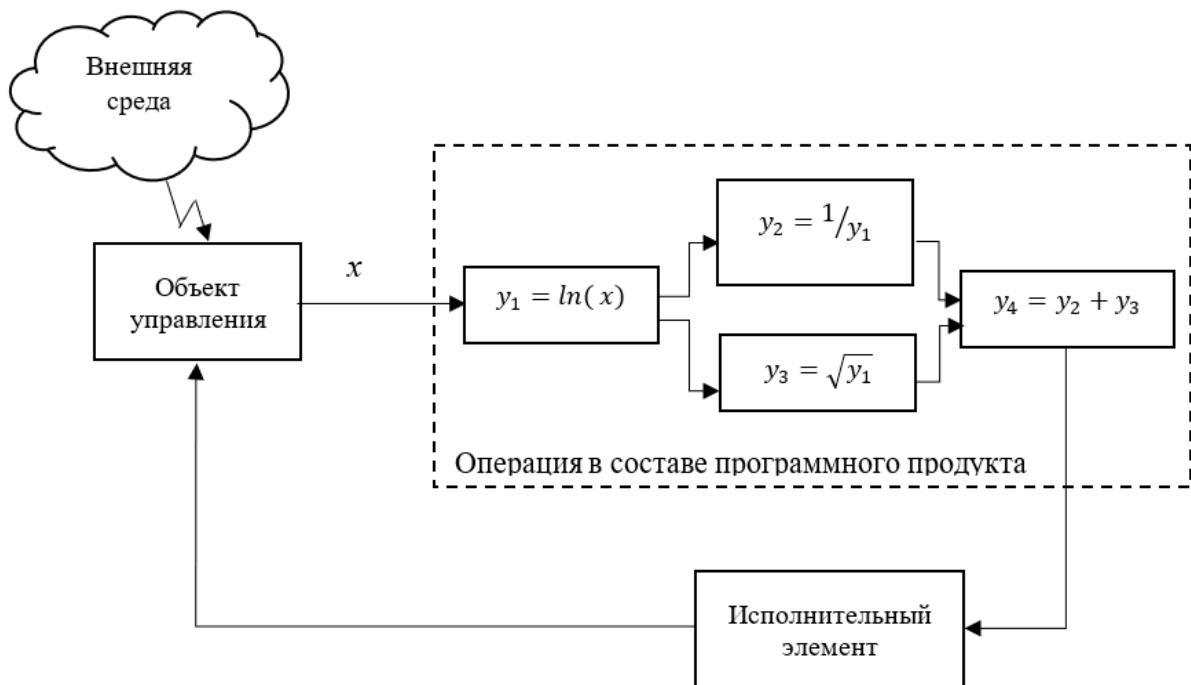


Рисунок 4.22 – Пример одной из операций, реализуемой программным продуктом в системе управления

Работоспособность программного продукта длительное время не вызывала нареканий. Однако в последующем два раза произошло аварийное прерывание вычислений.

В ходе исследований было установлено, что из-за изменения состояния внешней среды объекта управления область возможных значений  $x$  стала  $x \in (-1; 10)$ . Один из аварийных случаев связан с тем, что значение  $x$  оказалось  $x = 0$ . Другой – с тем, что  $x = 1$ . Схема устранения причин отказов свелась к тому, что в коды была добавлена проверка условия  $x \in (3; 5)$ . Если это условие не соблюдается, будет выдаваться сообщение «недопустимое значение аргумента» и вычисления не производятся.

*Использование схемы AFD-1:*

Шаг 1. Формулировка проблемы.

Имеется программный продукт, реализующий вычислительные операции:

$$y_1 = \ln(x); y_2 = 1/y_1; y_3 = \sqrt{y_1}; y_4 = y_2 + y_3. \quad \text{Безаварийное} \quad \text{завершение}$$

вычислений возможно при  $x > 1$ .

На этапе разработки предполагалось, что областью возможных значений аргумента всегда будут значения  $x \in (3; 5)$ , в силу чего в тексте программы отсутствовала проверка условия  $x > 1$ , что и послужило причиной аварийных завершений вычислений, так как один раз  $x$  приняло значение  $x = 0$ , один раз –  $x = 1$ . Снижение качества функционирования объекта управления падало, так как, хотя соблюдалось условие  $x > 1$ , в действительности значение  $x$  не всегда удовлетворяло условию  $3 < x < 5$ , а могло лежать в диапазоне  $-1 < x < 10$ .

Шаг 2. Описание идеального сценария  $S_0$ .

Идеальный сценарий состоит в получении значения  $x \in (3; 5)$  и проведении вычислений  $y_4$  в соответствии с выражением в шаге 1.

Шаг 3. Локализация области отказа.

Отказы были обусловлены тем, что отсутствовал контроль допустимости значений аргумента, т.е. не соблюдалось условие  $x > 1$ . Снижение качества функционирования объекта управления обусловлено тем, что при нарушении условия  $x \in (3; 5)$  не осуществлялся переход на иные алгоритмы расчета  $y_4$  гарантировавшие выполнение условия  $y_4 \in (y_4^{(4)}, y_4^{(6)})$ .

Иными словами, не реализовывалась схема:

$$y_4 = \begin{cases} A^{(1)} \text{ при } 1 < x \leq 3, \\ A^{(2)} \text{ при } 3 < x < 5, \\ A^{(3)} \text{ при } 5 \leq x < 10, \end{cases}$$

где  $A^{(i)}$  – алгоритм расчета  $y_4$  при нахождении  $x$  в  $i$ -м из указанных диапазонов. Алгоритм  $A^{(2)}$  совпадает со схемой расчета, представленной на рисунке 3.22.

Шаг 4. Формулирование расширенной инвертированной проблемы.

4.1. Непосредственная причина отказов программного продукта состояла в том, что в тексте программы отсутствовал контроль допустимости значений входных параметров. Снижение качества информационной поддержки управления

объектом было обусловлено тем, что не был предусмотрен переход к разным алгоритмам обработки данных в зависимости от того, какому диапазону принадлежит значение  $x$ .

4.2. При создании программного продукта необходимо так организовать процесс его производства, чтобы в готовом продукте как можно чаще наблюдались аварийные завершения вычислений и в случае безаварийного завершения вычислений результаты расчетов не соответствовали состоянию бизнес-процесса.

#### Шаг 5. Поиск решения инвертированной проблемы.

##### 5.1. Поиск очевидных решений.

а) В техническом задании описание бизнес-процесса, поддержку которого должен обеспечить программный продукт, привести в максимально общем виде, препятствующем определению допустимых значений  $x$ , при которых не происходит аварийных завершений вычислений.

б) Исключить в техническом задании требование перехода на разные схемы обработки данных в зависимости от свойств исходных данных. Оставить лишь описание схемы расчетов, соответствующих идеальному сценарию реализации бизнес-процесса.

##### 5.2. Идентификация ресурсов.

В рассматриваемом случае условиями (ресурсами), способствующими реализации инвертированной проблемы, являются:

- недостаточная формализация описания бизнес-процессов, обеспечивающих управление объектом;
- недостаточная формализация условий выбора схемы расчетов в зависимости от свойств исходных данных.

##### 5.3. Поиск способов получения ресурсов.

В рассматриваемом случае наибольшее число латентных дефектов можно обеспечить следующими способами:

- 1) поручить разработку технического задания представителям заказчика и исключить из этого процесса представителей исполнителя, обеспечив возможность недостаточно формального и детального (с точки зрения разработки программного продукта) описания особенностей бизнес-процессов;

2) привлечь к разработке программного продукта специалистов, не владеющих в достаточной степени базовыми приемами разработки и испытания программных продуктов;

3) выделить на реализацию программного проекта ресурсов (времени разработки), заведомо недостаточных для обеспечения качественной разработки и исследования функциональных и нефункциональных свойств программного продукта.

Шаг 6. Формирование гипотезы по способам реализации инвертированной проблемы.

Можно утверждать, что основной способ реализации инвертированной проблемы – обеспечение низкого качества управления программным проектом. Причиной возникновения латентных дефектов на разных стадиях жизненного цикла программных продуктов являются системные ошибки в организации программных проектов и различного рода персонифицированные ошибки, допускаемые разработчиками в силу того, что программные продукты относятся к классу субъектоцентрических систем.

Шаг 7. Действия, препятствующие реализации инвертированной проблемы.

Формирование барьеров (организационных, технологических), исключающих возможность возникновения условий (получение ресурсов), необходимых для реализации расширенной инвертированной проблемы.

1) Организационные барьеры: разработка технического задания коллективом, в состав которого входят компетентные в своих профессиональных областях представители заказчика и исполнителя; контроль наличия в техническом задании формальных моделей управления объектом и моделей информационной поддержки управления.

2) Технологические барьеры: исследование поведения программного продукта при непредсказуемых изменениях входного параметра (например, посредством *ad-hoc* тестирования); контроль значения входного параметра  $x$  и выбор схемы расчета  $y_4$  с учетом того, какому из диапазонов принадлежит измеренное значение  $x$  (шаг 3).

Таким образом, в целях всестороннего анализа функциональной надёжности ИС на предпроектной стадии предложена схема событийного анализа в рамках подхода *AFD*, в виде систематической процедуры для выявления уязвимых мест в ИС (т.е. выявление мест, в которых целесообразно размещение «барьеров»). Ограничениями *AFD* с точки зрения управления функциональной надежностью ИС является ориентация на статические модели, исключающая возможность учёта временной динамики процессов.

## **Выводы по главе 4**

1) Предложен метод построения многомерных зависимостей на основе совместного использования измерительных данных и экспертных оценок, основу метода составляет преобразование исходных данных, изначально представляемых в разной форме, к виду закона распределения непрерывной случайной величины. В предложенном методе многосвязный объект описывается совокупностью одномерных строгих непараметрических регрессионных зависимостей.

Предложенный подход, позволяет исследовать поведение многосвязных объектов в случае, когда некоторые компоненты вектора независимой переменной являются ненаблюдаемыми, но им ставятся в соответствие экспертные оценки.

2) Предложен метод оценки состояния функциональной надежности информационных сервисов с учетом субъективных оценок пользователей, основанный на концепции профилей. Основу метода составляет преобразование различных по форме исходных данных к единому виду. Данный подход позволяет получать метрические оценки надежности информационных сервисов на основе совокупного использования как измерительных данных, так и субъективных оценок пользователей. Практической направленностью предлагаемого метода является определение приоритетных направлений совершенствования функциональной надежности.

3) Предложен метод структурного моделирования отказов, основанный на совместном использовании аппарата схем сопряжения и таблиц истинности. Это создает основу для построения структурных моделей, обеспечивающих моделирование отказов в рамках как нисходящего, так и восходящего подходов к исследованию. Предложенный метод создает информационную основу для сравнения альтернативных вариантов проектных решений по критерию функциональной надежности ИС на предпроектной стадии.

4) С целью разноспектрного анализа функциональной надежности ИС на предпроектной стадии предложена схема анализа событий в рамках AFD в виде систематической процедуры для выявления уязвимых мест в ИС (т.е. выявление мест, в которых целесообразно устанавливать «барьеры»).

Основные результаты, представленные в четвертой главе, опубликованы в статьях автора данного исследования в рецензируемых печатных изданиях из перечня, утвержденного ВАК [8, 11, 15, 17], в международных изданиях, индексируемых *SCOPUS* [155, 156, 136] и в материалах международных конференций [12, 132].

## ГЛАВА 5. РАЗРАБОТКА ДЕСКРИПТИВНЫХ МОДЕЛЕЙ БАЗОВЫХ ХАРАКТЕРИСТИК ПРОЕКТА РАЗРАБОТКИ ИНФОРМАЦИОННЫХ СИСТЕМ

Данная глава посвящена разработке дескриптивных моделей базовых характеристик проекта разработки ИС на основе математико-статистических методов и системных архетипов.

Первый параграф настоящей главы посвящен разработке модели проекта создания ИС, основу построения модели проекта составляет рассмотрение его как многосвязного объекта управления. Эмпирические функциональные зависимости соответствуют прямым и перекрестным связям многосвязного объекта.

Во втором параграфе рассматривается формирование параметров модели управления проектом на основе линеаризации функциональных зависимостей.

В третьем параграфе рассмотрен вопрос оценки бюджета проекта по критериям удовлетворённости акторов.

В четвертом параграфе рассматриваются вопросы построения моделей ситуаций, возникающих при реализации проектов. Рассмотрены типовые ситуации на основе системных архетипов, а также контурного представления.

В пятом параграфе представлен подход к оценке согласованности мнений неоднородных акторов. В качестве инструмента оценки согласованности мнений неоднородных акторов на предпроектной стадии используются системные архетипы, модель «Дом качества» и коэффициенты корреляции.

## 5.1 Модель проекта разработки информационных систем

Получившие в настоящее время широкое распространение методы и модели управления проектами – ISO 21500 [166], ICB IPMA [163], PMBOK PMI [101, 127], НТК СОВНЕТ [106] предназначены для уровня исполнителей: руководителей проектов, управляющей команды, специалистов офисов. Недостаточно развита методическая и модельная основа поддержки деятельности основных заинтересованных сторон, т.е. уровня принятия стратегических решений.

В [41] подчеркивается, что от обоснованности стратегических решений зависит 50% успеха проектной деятельности. Там же отмечается, что к настоящему времени наибольшее распространение среди подходов к поддержке принятия стратегических решений с учетом позиций разных заинтересованных сторон получили модели, реализованные в рамках дескриптивного подхода, ограничивающиеся исследованиями на содержательно-описательном уровне. Подчеркивается необходимость совершенствования обеспечения на основе математических моделей, в том числе моделей со стохастическими параметрами.

Формирование консолидированного мнения неоднородных акторов<sup>13</sup> в условиях все возрастающей сложности, неопределенности, скорости изменения состояния окружающей среды и различного влияния этих изменений на их представление о ценностях и удовлетворенность своим состоянием требует наличия инструментов, позволяющих:

- получать хорошо интерпретируемые и одинаково понимаемые акторами оценки основных характеристик проекта;
- учитывать исторический опыт реализации аналогичных проектов;

---

<sup>13</sup> Под неоднородными акторами понимается субъект, имеющий ресурсы, полезные в той ситуации, в которой оказывается актор, за счет чего способный воздействовать на ситуацию. Вместе с тем, ресурсов отдельного актора недостаточно для того, чтобы урегулировать проблемную ситуацию. Актор имеет свою точку зрения окружающую действительность, и в этом смысле акторы являются неоднородными. Оказавшись связанными общей проблемной ситуацией, акторы осознают её по-разному, вместе с тем понимают необходимость согласованных действий, и готовые нести ответственность за последствия принимаемых решений [34].

– использовать как фактические (исторические) данные, связанные с реализацией аналогичных проектов, так и нематериальные активы акторов (аксиологические знания).

Одной из наиболее известных концептуальных моделей проекта, соответствующих начальной фазе жизненного цикла, является «Треугольник проекта» [144]. Эта модель определяет базовые характеристики, составляющие основу построения систем управления проектами.

Вместе с тем, необходимо обратить внимание на то, что в литературных источниках подчеркивается важность учета представлений о ценностях и удовлетворенности результатами проекта лишь одной группы акторов – потребителей. Вне поля зрения незаслуженно остаются интересы второго участника проекта – исполнителя. В то же время очевидно, что ход проекта и потребительские свойства результатов проекта, в том числе напрямую зависят от отношения к нему исполнителей.

Оценка стоимости продукта является отправной точкой определения бюджета проекта. Наибольшую неопределенность представляет оценка стоимости программных компонентов ИС, т.к. реализация программных проектов основана на использовании нематериальных активов – интеллектуальных ресурсов субъектов, относящихся к категориям пользователей и разработчиков.

Основу предлагаемого подхода к решению задачи составляют следующие допущения:

– Потребительские свойства конечного продукта (ИС) определяются организацией и ходом реализации проекта. Основанием этому служат, во-первых, известные утверждения о том, что поведение системы определяются ее устройством [84], во-вторых, известное утверждение о том, что причинами нежелательных событий, возникающих при управлении сложными субъектоцентрическими системами, в первую очередь являются ошибки в организации управления сложной системой, и лишь во-вторую очередь, ошибки операторов [203].

– Проект создания сложной системы рассматривается как многосвязный объект управления. Это дает основание рассматривать прямые и перекрестные связи между входными и выходными параметрами проекта как строгие функциональные зависимости с неизменной структурой и параметрами. Содержание этого допущения является выражением известного положения о взаимовлиянии качества управления проектом и потребительских свойств получаемого продукта (см., например,[121]).

– ИС относятся к классу субъектоцентрических систем. В силу этого при оценивании проектов нужно одновременно использовать как их измерительные данные, так и субъективные оценки потребителей и исполнителей. При оценивании качества хода проекта и потребительских свойств получаемого продукта в равной степени важно учитывать удовлетворенность исполнителей и заказчиков [113].

– В качестве входных параметров проекта рассматриваются бюджет и длительность реализации проекта. В качестве выходных – удовлетворенность представителей заказчика и исполнителей проекта. Содержание этого допущения является конкретизацией компонентов известной системной модели «Треугольник проекта» в редакции 2015 года [144].

– Ожидаемый исход проекта в равной мере определяется как ограничениями на бюджет и длительность реализации проекта, так и опытом исполнителя в реализации проектов аналогичного содержания [144, 145]. Содержание этого допущения является выражением ресурсного подхода с одной стороны и того обстоятельства, что проект сложной системы является разновидностью субъектоцентрических систем [70].

– Допущение о сопоставимости и однородности (в статистическом смысле) исторических данных об основных параметрах проектов и полученных в ходе их реализации продуктов. Основанием для такого допущение является, во-первых, то, что организации-разработчики специализируется на реализации проектов в определенной предметной области, во-вторых, то, что внутри организации накапливается опыт решения задач определенной тематической направленности.

Проекту как многосвязному объекту управления может быть поставлена в соответствие модель, представленная на рисунке 5.1.

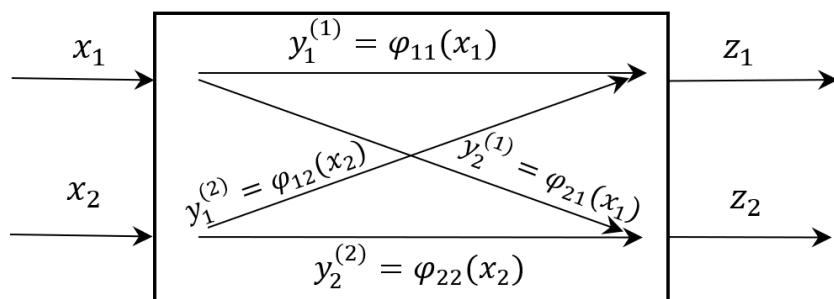


Рисунок 5.1 – Модель проекта как многосвязного объекта управления

Входными (управляемыми) параметрами модели являются:

$x_1$  – бюджет проекта;

$x_2$  – длительность реализации проекта.

Выходными параметрами являются:

$z_1$  – удовлетворенность заказчика;

$z_2$  – удовлетворенность исполнителя.

Постулируется положение о том, что эмпирические функциональные зависимости  $y_j^{(i)} = \varphi_{ij}(x_i)$ , ( $i, j = 1, 2$ ), соответствующие прямым и перекрестным связям многосвязного объекта, являются строгими.

В силу субъектоцентрической природы системы управления проектами, основанием построения формульных описаний прямых и перекрестных связей являются экспертные оценки субъектов, ранее привлекавшихся к реализации аналогичных проектов, а также исторические данные о параметрах ранее реализованных проектов. При построении формальных описаний прямых и перекрестных связей превалируют эмпирические и полуэмпирические математические модели.

Эмпирические функциональные зависимости строятся на основе измерительных исторических данных и экспертных оценок относительно потребительских свойств созданных продуктов и хода ранее реализованных проектов.

Совместное влияние входных характеристик  $x_i$  на выходные определяется на основе функциональных зависимостей, характеризующих прямые и перекрестные связи  $y_j^{(i)} = \varphi_{ij}(x_i)$ :

$$A^{(j)}: \{y_j^{(i)} = \varphi_{ij}(x_i)\} \rightarrow z_j = \Phi_j(x_1, x_2), i, j = 1, 2 \quad (5.1)$$

Здесь  $A^{(j)}$  – оператор свертки, соответствующий  $j$ -му выходному параметру модели.

Основу формирования операторов свертки  $A^{(j)}$ , преобразующих функциональные зависимости  $\{y_j^{(i)} = \varphi_{ij}(x_i)\}$ , характеризующие прямые и перекрестные связи в зависимости вида  $z_j = \Phi_j(x_1, x_2), i, j = 1, 2$ , составляют следующие положения.

1) Областью допустимых значений для характеристики удовлетворенности  $z_j$  будем считать интервал  $z_j \in [0,1], j = 1, 2$ . Нижняя граница интервала соответствует варианту, когда субъекты (потребитель/исполнитель) абсолютно не удовлетворены результатами /ходом проекта. Верхняя граница интервала соответствует их абсолютной удовлетворенности.

2) Удовлетворенность потребителя уменьшается по мере увеличения как бюджета  $x_1$ , так и длительности реализации проекта  $x_2$ . Поэтому функциональные зависимости  $y_j^{(i)} = \varphi_{ij}(x_i)$  являются обратными. При этом возможна компенсация одного параметра другим. К примеру, компенсация изменения длительности реализации проекта за счет изменения бюджета проекта и наоборот.

3) С точки зрения заказчика возможна реализация проекта нулевой длительности, что фактически соответствует приобретению готового продукта. Исходя из вышеизложенного можно утверждать, что  $\Phi_1(x_1, x_2)$  можно представить аддитивной функцией:

$$z_1 = \varphi_{11}(x_1) + \varphi_{21}(x_2) \quad (5.2)$$

4) Удовлетворенность исполнителей, напротив, растет с увеличением как бюджета  $x_1$ , так и длительности реализации  $x_2$  проекта. Постулируется, что исполнитель отказывается от реализации проекта, если хотя бы одна из его характеристик  $x_i (i=1,2)$  равна нулю.

Исходя из этих соображений можно заключить, что в качестве  $\Phi_2(x_1, x_2)$  можно использовать мультипликативную функцию:

$$z_2 = \varphi_{12}(x_1) * \varphi_{22}(x_2) \quad (5.3)$$

Таким образом, предложена модель проекта как многосвязного объекта управления. Входными параметрами модели являются бюджет и длительность реализации проекта. Выходными параметрами являются удовлетворенность заказчика результатами проекта и удовлетворенность исполнителя ходом проекта. Формирование сбалансированной системы характеристик проекта является одним из решающих факторов успешной реализации проекта.

## **5.2 Формирование параметров модели управления проектом на основе линеаризации функциональных зависимостей**

В данном разделе рассмотрим задачу оценки бюджета проекта в условиях ограничений на длительность реализации проекта, при котором достигается сбалансированность интересов заказчика и исполнителя, т.е. равный уровень удовлетворенности заказчика и исполнителя.

*Постановка и схема решения задачи:*

*Известен:* вид функциональных зависимостей  $y_j^{(i)} = \varphi_{ij}(x_i)$ ,  $i,j=1, 2$ .

*Требуется:* оценить значения параметров проекта  $x_i^{(OUT)}$  ( $i=1,2$ ), при которых достигается наименьшее различие в удовлетворенности результатами проекта потребителей  $y_1$  и исполнителей  $y_2$ .

Очевидно, что наименьшее различие в удовлетворенности результатами проекта потребителями и ходом проекта исполнителями достигается при  $y_1=y_2$ . Этому условию соответствует соотношение:

$$\varphi_{11}(x_1) + \varphi_{21}(x_2) = \varphi_{12}(x_1) * \varphi_{22}(x_2) \quad (5.4)$$

$$0 \leq \varphi_{11}(x_1) + \varphi_{21}(x_2) \leq 1 \quad (5.5)$$

$$0 \leq \varphi_{12}(x_1) * \varphi_{22}(x_2) \leq 1 \quad (5.6)$$

$$x_i \geq 0, i = 1, 2 \quad (5.7)$$

Рассмотрим случай, когда прямые и перекрестные связи представляются линейными зависимостями. При этом критичность величины бюджета и длительности реализации проекта для заказчика характеризуются параметрами  $a_{11}$  и  $a_{21}$ , а предыдущий опыт реализации проектов исполнителями характеризуется параметрами  $b_{12}$  и  $b_{22}$ .

Совокупное влияние  $x_1$ ,  $x_2$  на  $y_1$ ,  $y_2$  определяются соотношениями:

$$z_1 = (a_{01} - a_{11}x_1) + (a_{02} - a_{21}x_2) \quad (5.8)$$

$$z_2 = b_{12}x_1 * b_{22}x_2 \quad (5.9)$$

Предполагая равное влияние бюджета и длительности реализации проекта на удовлетворенность заказчика, а также то, что при  $x_1=x_2=0$  результат получается моментально и даром) удовлетворенность заказчика максимальна, получаем  $a_{01}=a_{02}=0.5$ . Или:

$$z_1 = (a_{01} - a_{11}x_1) + (a_{02} - a_{21}x_2) = 1 - a_{11}x_1 - a_{21}x_2 \quad (5.10)$$

$$z_2 = b_{12}x_1 * b_{22}x_2 = Bx_1x_2 \quad (5.11)$$

С учетом того, что  $y_j \leq 1$ , из (9) получаем:

$$x_2 \geq \frac{1 - a_{11}x_1}{a_{21}} = C - e x_1 \quad (5.12)$$

где  $C = \frac{1}{a_{21}}$ ;  $e = \frac{a_{11}}{a_{21}}$ .

Из (5.11) получаем:

$$x_2 \leq \frac{k}{x_1} \quad (5.13)$$

где  $k = \frac{1}{B}$ .

Соотношения (5.12) и (5.13) определяют область, в которой возможен поиск  $x_1^{(OUT)}$  и  $x_2^{(OUT)}$ . Эта область на рисунке 5.2 выделена штриховкой.

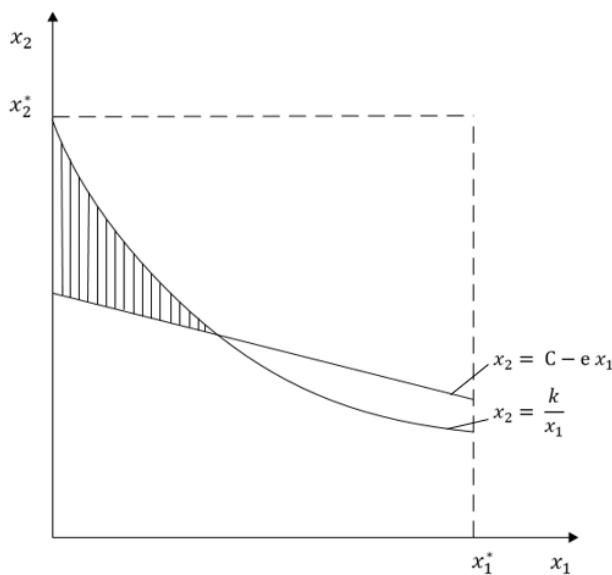


Рисунок 5.2 – Область поиска  $x_i^{(OUT)}$ ,  $i = 1, 2$

На приведенных ниже рисунках 5.3, 5.4 представлены пример результатов, соответствующие различным представлениям о ценностях заказчика и исполнителя.

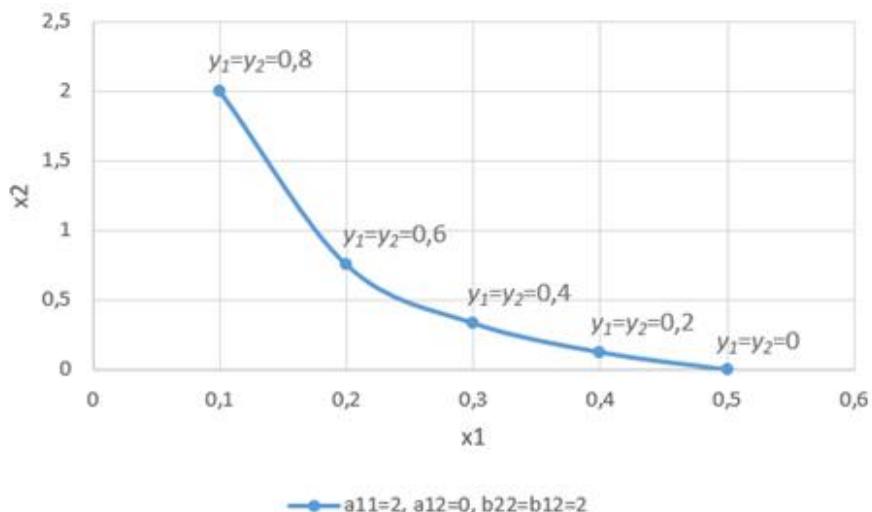


Рисунок 5.3 – Удовлетворенность заказчика определяется длительностью реализации проекта и практически не зависит от величины бюджета проекта.

Заданные значения параметров (рисунок 5.3) соответствуют случаю, когда удовлетворенность заказчика определяется длительностью реализации проекта и практически не зависит от величины бюджета проекта.

Подобная ситуация может иметь место в случае необходимости обеспечения информационной поддержки критически важных задач в условиях жестких ограничений на время обеспечения информационной поддержки.

Удовлетворенность исполнителя определяется как величиной бюджета проекта, так и длительностью реализации проекта. Содержательно такая ситуация, например, может иметь место при обеспечении информационной поддержки ликвидации последствий чрезвычайной ситуации, когда в качестве заказчика на информационное обслуживание выступают органы исполнительной власти, а исполнителями – научно-производственные организации, являющиеся структурными подразделениями этих органов.

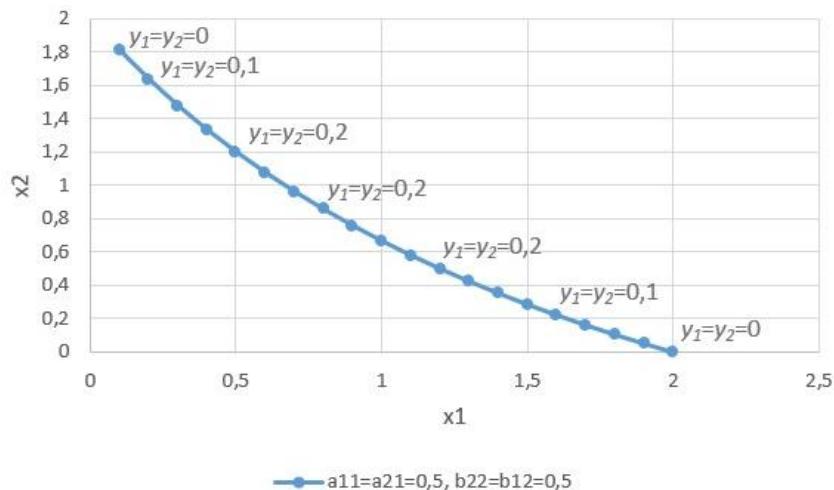


Рисунок 5.4 – Равное влияние бюджета и длительности реализации проекта на удовлетворенность заказчика и исполнителя

На основании полученных зависимостей с учетом реальных ограничений на бюджет  $x_1^*$  и длительность реализации проекта  $x_2^*$  на ранних стадиях проекта можно оценить предполагаемое качество результатов (выражаемое степенью удовлетворенности) с учетом прежнего опыта взаимодействия исполнителя и заказчика. Предлагаемый подход позволяет повысить обоснованность решений о целесообразности реализации проекта силами предполагаемого исполнителя с учетом приоритетности бюджета и длительности реализации проекта для заказчика. Ограничениям модели является: линейный характер функциональных зависимостей, характеризующих прямые и перекрестные связи.

### 5.3 Оценка бюджета проекта по критериям удовлетворённости акторов

Бюджет проекта преимущественно формируется в процессе переговоров между заказчиком и исполнителем. Разработка формализованных методик, направленных на информационное обеспечение поиска консолидированных решений заинтересованными сторонами в отношении оценки стоимости результатов проекта, длительное время находится в сфере внимания специалистов, занимающихся разработкой ИС. Оценке стоимости программных продуктов и проектов посвящены работы [66, 81, 83, 124, 125]. Подходы к оценке стоимости, изложенные в этих работах, могут быть классифицированы следующим образом.

- Слабо формализованные методы. Их областью применимости является начальная стадия проекта. Определение стоимости предполагаемого продукта основано на эмпирических данных, относящихся к ранее реализованным проектам.
- Методы, основанные на анализе архитектуры программных продуктов. Наиболее известным представителем методов, относящихся к этому классу, является метод функциональных точек (*Function Points - FP*). Метод инвариантен к языку программирования и среде разработки. Существенным ограничением методов этого класса является предположение о высоком качестве функциональных требований [66].
- Методы, основанные на подсчёте числа строк кода (*Source Lines of Code*). Основу этих методов составляют эмпирические модели, устанавливающие размер затрат в зависимости от числа строк кода и характеристик среды реализации проекта. Ограничениями этого подхода являются низкая достоверность оценивания числа строк кода на начальной стадии проекта.

В бюджете проекта наибольшую неопределенность представляет оценка стоимости программных компонентов ИС, т.к. реализация программных проектов основана на использовании нематериальных активов – интеллектуальных ресурсов акторов.

В данном разделе рассмотрим задачу оценки бюджета проекта в условиях различных значений уровней удовлетворенности заказчика и исполнителя.

Основу оценивания возможности реализации проекта составляет нахождение значений  $x_1, x_2$ , при которых уровень удовлетворенности заказчика и исполнителя не ниже некоторых заданных значений, т.е.

$$\alpha \leq \Phi_1(x_1, x_2) \leq 1 \quad (5.14)$$

$$\beta \leq \Phi_2(x_1, x_2) \leq 1 \quad (5.15)$$

Качественные графические модели для (5.14) и (5.15) представлены на рисунке 5.5 и рисунке 5.6.

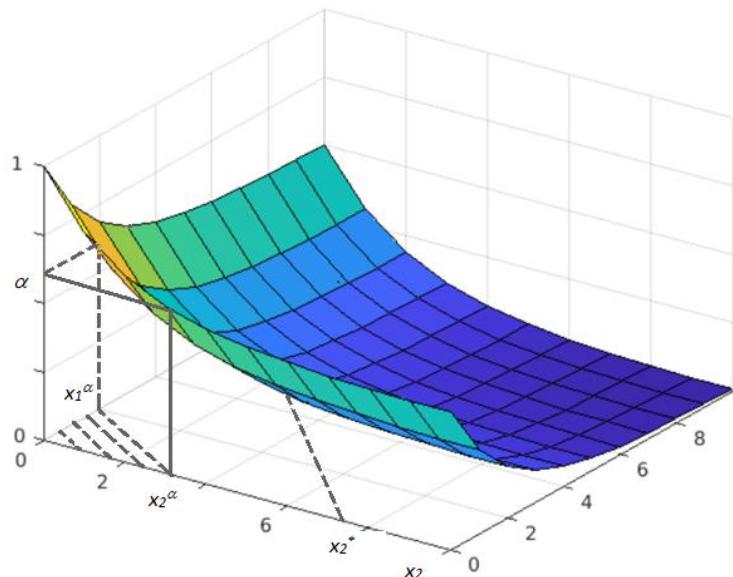


Рисунок 5.5 – Графическая модель  $\Phi_1(x_1, x_2)$

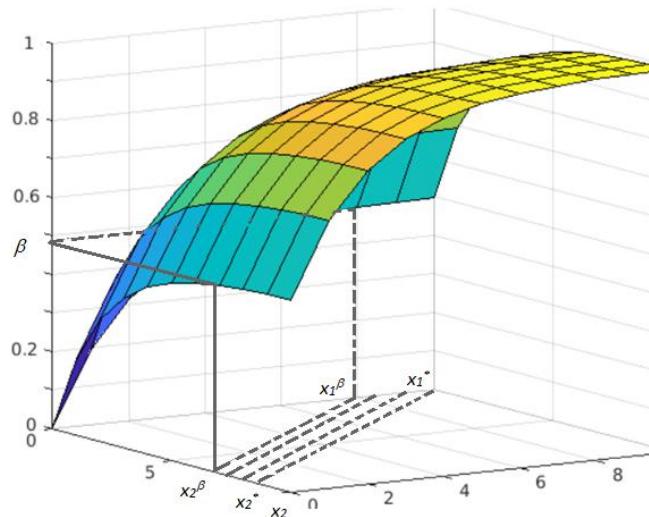


Рисунок 5.6 – Графическая модель  $\Phi_2(x_1, x_2)$

На рисунках 5.5 и 5.6 обозначены:

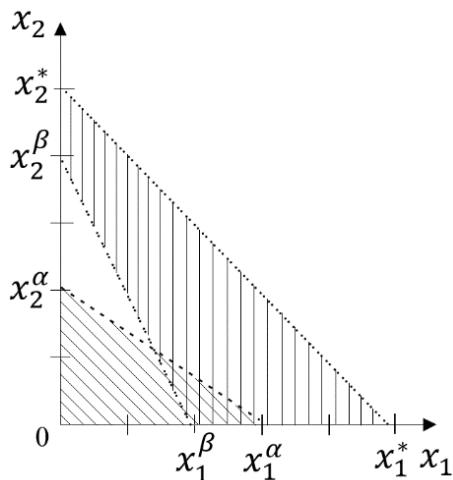
$x_i^*$  – граничные значения параметров бюджета и длительности реализации, при повышении которых реализация проекта теряет смысл;

$x_i^\alpha$  – предельные (верхнее) значения параметров проекта, при повышении которых удовлетворенность заказчика уменьшается ниже желаемого для него значения  $\alpha$ ;

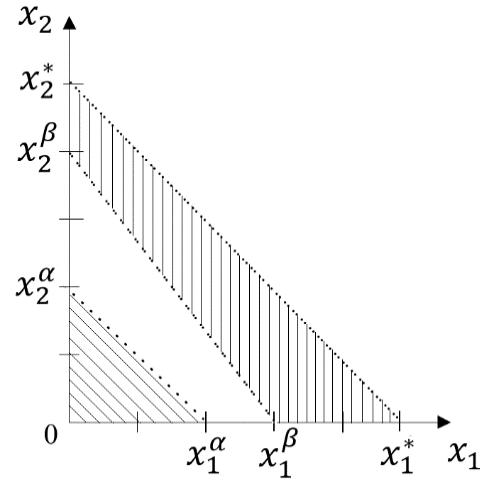
$x_i^\beta$  – предельные (нижние) значения параметров проекта, при уменьшении которых удовлетворенность исполнителя оказывается ниже желаемого для него значения  $\beta$ .

Пунктирными линиями на рисунках выделены области допустимых решений;

На рисунке 5.7 показано графическое представление нахождения консолидированного решения между потребителем/заказчиком и исполнителями.



(а) – возможность нахождения сбалансированного решения с учетом особенностей  $\varphi_{ij}(x_i)$ , а также значений  $\alpha$  и  $\beta$



(б) – невозможность нахождения сбалансированного решения с учетом особенностей  $\varphi_{ij}(x_i)$ , а также значений  $\alpha$  и  $\beta$

Рисунок 5.7 – Графическое представление нахождения консолидированного решения между потребителем/заказчиком и исполнителями

Таким образом, реализация проекта возможна лишь в условиях, во-первых, сбалансированности запросов к параметрам проекта заказчика и исполнителя (что выражается значениями  $\alpha$  и  $\beta$ ).

Во-вторых, сопоставимости персонального опыта как заказчика, так и исполнители, накопленного ими при реализации прежних проектов (что выражается посредством функциональных зависимостей  $\varphi_{ij}(x_i)$ ,  $i, j = 1, 2$ ).

В качестве структур эмпирических зависимостей могут быть выбраны следующие:

$$\varphi_{i1}(x_i) = e^{-\lambda_i x_i} \quad (5.16)$$

$$\varphi_{i2}(x_i) = (1 - e^{-q_i x_i}), i = 1, 2 \quad (5.17)$$

Эти зависимости удовлетворяют всем ограничениям, накладываемым на  $\Phi_j(x_1, x_2)$  и формируемым посредством (5.2) и (5.3). Особенностью оценивания параметров эмпирических функциональных зависимостей является:

- 1) параметры  $x_1, x_2$  являются метрическими характеристиками, а степень удовлетворенности субъектов выражается вербальными оценками;
- 2) различные целевые группы пользователей/представителей заказчика могут выражать разные уровни удовлетворенности потребительскими свойствами ИС, так же, как и различные группы исполнителей, задействованные в решении различных классов задач, могут выражать разные уровни удовлетворенности ходом проекта.

Иными словами, отдельному проекту изначально ставится в соответствие две метрические характеристики  $x_1, x_2$  и различное множество вербальных оценок по каждому из проектов различных групп потребителей и исполнителей.

Поэтому первым этапом определения параметров эмпирических зависимостей является обеспечение равенства объемов параметров  $\{x_i\}_1^N, (i = 1, 2); \{z_j\}_1^N, (j = 1, 2)$ .

В [14] подробно описана процедура квантификации экспертных оценок, даваемых субъектами, принадлежащими к различным подгруппам, принадлежащих группам пользователей и исполнителей. Множество оценок субъектов по каждому из проектов сводится к скаляру  $z_j^{(l)}$ .

В результате применения процедуры каждому  $i$ -му проекту ставится в соответствие интегральные оценки удовлетворенности пользователей/заказчика  $z_1^{(j)}$  и исполнителей  $z_2^{(j)}$ .

В результате расчет параметров эмпирических моделей (5.16) и (5.17) сводится к решению задач оптимизации без ограничений.

Для модели (5.16):

$$\sum_{i=1}^N (z_1^{(j)} - \sum_{k=1}^2 e^{-\lambda_k x_k^{(i)}}) \underset{\lambda_1, \lambda_2}{\longrightarrow} \min \quad (5.18)$$

Для модели (5.17):

$$\sum_{i=1}^N (z_2^{(j)} - \prod_{k=1}^2 (1 - e^{-q_k x_k^{(i)}})) \underset{q_1, q_2}{\longrightarrow} \min \quad (5.18)$$

$N$  – число однотипных проектов, схожих с тем, параметры которого оцениваются.

Выбор квадратичной целевой функции обусловлен высокой устойчивостью полученных оценок. Особенности решения задач, подобных (5.18) и (5.19) обсуждается в [56].

Параметры моделей  $\lambda_k, q_k$  ( $k = 1, 2$ ) содержательно характеризуют опыт пользователей/заказчика и исполнителей, полученный при реализации предыдущих проектов. Полученные значения  $\lambda_k, q_k$  ( $k = 1, 2$ ) позволяют оценить возможность нахождения приемлемых решений с учетом ограничений  $x_i^*$  ( $i = 1, 2$ ), а также желаемых значений  $\alpha$  и  $\beta$  (см. раздел 5.2).

Предлагаемый подход к оцениванию одной из основных характеристик проекта (бюджета) основан на эмпирических моделях, основу построения которых составляет опыт реализации предыдущих проектов. Такой подход отражает субъектоцентрическую природу проектов. Принципиальным отличием предлагаемого подхода от других подходов к оценке бюджета проекта на основе исторических данных является ориентация не только на трудоемкость разработки, но и на приемлемый как для потребителя, так и для исполнителей уровень удовлетворенности.

Формализация процедуры нахождения области возможных решений ускоряет процесс формирования консолидированного решения при определении базовых характеристик проекта.

## 5.4 Построение моделей ситуаций, возникающих при реализации проектов

В литературе по управлению проектами [4, 30, 81, 100, 113, 124, 142] большое внимание уделяется вопросам, связанным с урегулированием ситуаций, обусловленных субъектоцентрической природой проектов. Неизбежность возникновения ситуаций, требующих урегулирования находит выражение в известной модели «конус неопределенности проекта». Неопределенность сред реализации проектов является одним из факторов совершения правообладателями организационных ошибок, имеющих наиболее тяжелые последствия с точки зрения функциональной надежности.

Описанные в литературе рекомендации по урегулированию ситуаций носят качественный характер, недостаточное развитие получили структурные и математические модели, посредством которых можно обеспечить информационную поддержку выбора обоснованного подхода к урегулированию ситуаций с учетом их природы.

В настоящем разделе предлагается подход, ориентированный на построение математических моделей, с разных сторон описывающих типовые ситуации, возникающие при реализации проектов создания ИС.

Укрупненная схема построения моделей, предназначенных для обеспечения информационной поддержки урегулирования ситуаций, возникающих при управлении проектами разработки ИС, представлена на рисунке 5.8.

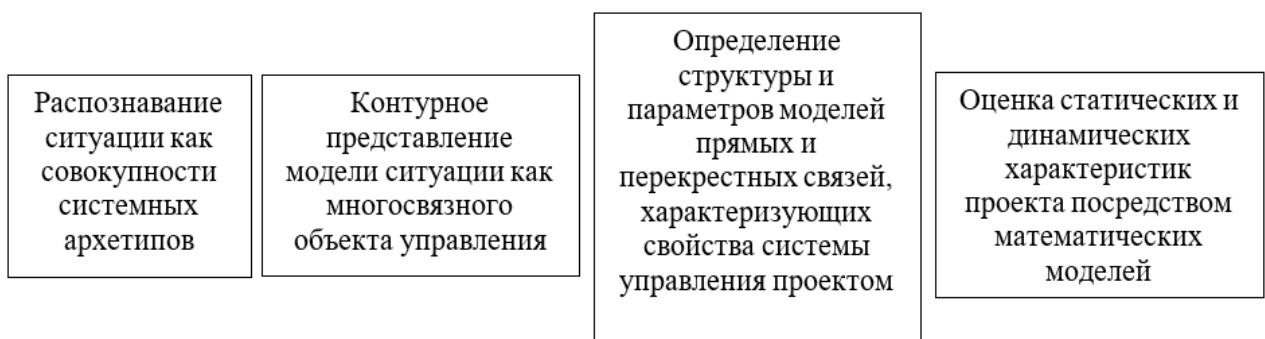


Рисунок 5.8 – Укрупненная схема построения моделей

Первой задачей в урегулировании ситуации является определение её содержания. В качестве подхода к решению этой задачи целесообразно использовать совокупность системных архетипов (содержание системных архетипов подробно рассмотрено в разделе 2.5).

Следующим шагом после выделения проблемной ситуации, связанной с управлением свойствами ИС, является построение структурных моделей ситуаций, соответствующих выделенным архетипам, в виде многосвязного объекта управления. В общем случае структурная модель имеет вид, представленный ниже на рисунке 5.9.

Входными параметрами модели являются – факторы ситуации.

Выходными – симптомы ситуации.

Структура и параметры прямых и перекрестных связей, характеризуют влияние внешних факторов на симптомы, и являются отображением структуры объекта.

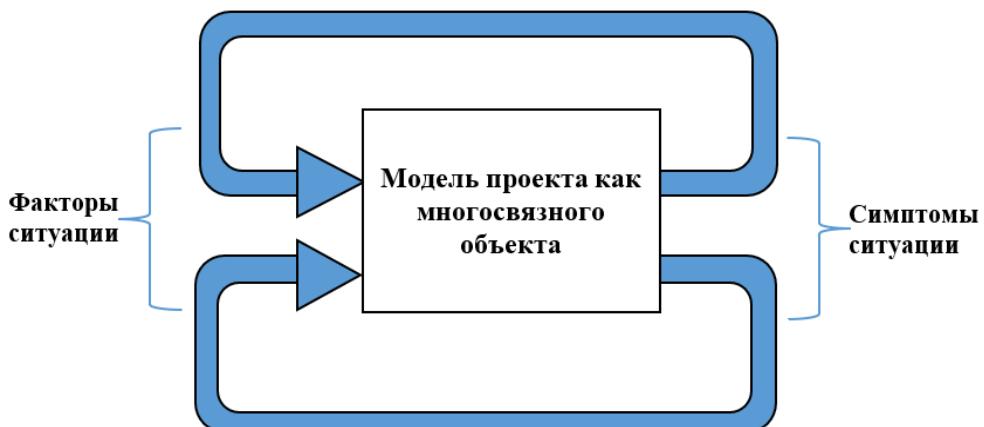


Рисунок 5.9 – Структурная модель ситуации в виде многосвязного объекта управления

Вследствие того, что на разных этапах жизненного цикла ИС локальные цели, задачи и подходы к решению задач различаются, что вынуждает соответствующим образом изменять организацию системы управления, можно утверждать, что структура и параметры прямых и перекрестных связей также изменяются. Особенности формирования прямых и обратных связей в контурных моделях формируют эмерджентные свойства многосвязного объекта.

*Рассмотрим пример описания типичных для средних проектов ситуаций посредством системных архетипов, а также контурного представления системных архетипов.*

Количественная характеристика понятия «средний программный проект» приведена в [154].

Возвращаясь к вопросу о различиях в подходах к оценкам проекта заказчиком и исполнителями необходимо отметить следующее.

1) Удовлетворенность потребителя результатами проекта, а исполнителя – организацией и ходом реализации проекта являются равнозначными факторами, определяющими успех проекта.

Удовлетворенность потребителя является интегральной характеристикой соответствия потребительских свойств ИС представлениям о ценностях потребителей. Удовлетворенность исполнителя является интегральной характеристикой соответствия внутреннего устройства проекта (организационного, финансового, технологического на всех стадиях его жизненного цикла) ожидавшемуся на предпроектной стадии.

2) С точки зрения ресурсного обеспечения (интегрально выражаемого бюджетом проекта) потребители стремятся сократить величину бюджета, исполнители же – увеличить, хотя исполнители осознают неизбежность наличия ограничений на предлагаемую потребителем величину бюджета.

3) Новые возможности, предоставляемые системами сетецентрического управления, стимулируют рост ценностей, получаемых потребителями, что, в свою очередь, стимулирует рост заинтересованности в развитии ИС. Увеличение масштабов и сложности динамических объектов управления вынуждает потребителей постоянно вкладывать ресурсы в модернизацию и развитие ИС.

4) Возрастающая сложность запросов со стороны потребителей способствует развитию теоретической, методической и инструментальной базы создания информационных и киберфизических систем.

Этой ситуации можно поставить в соответствие системный архетип «Случайные противники» (показано на рисунке 5.10.).

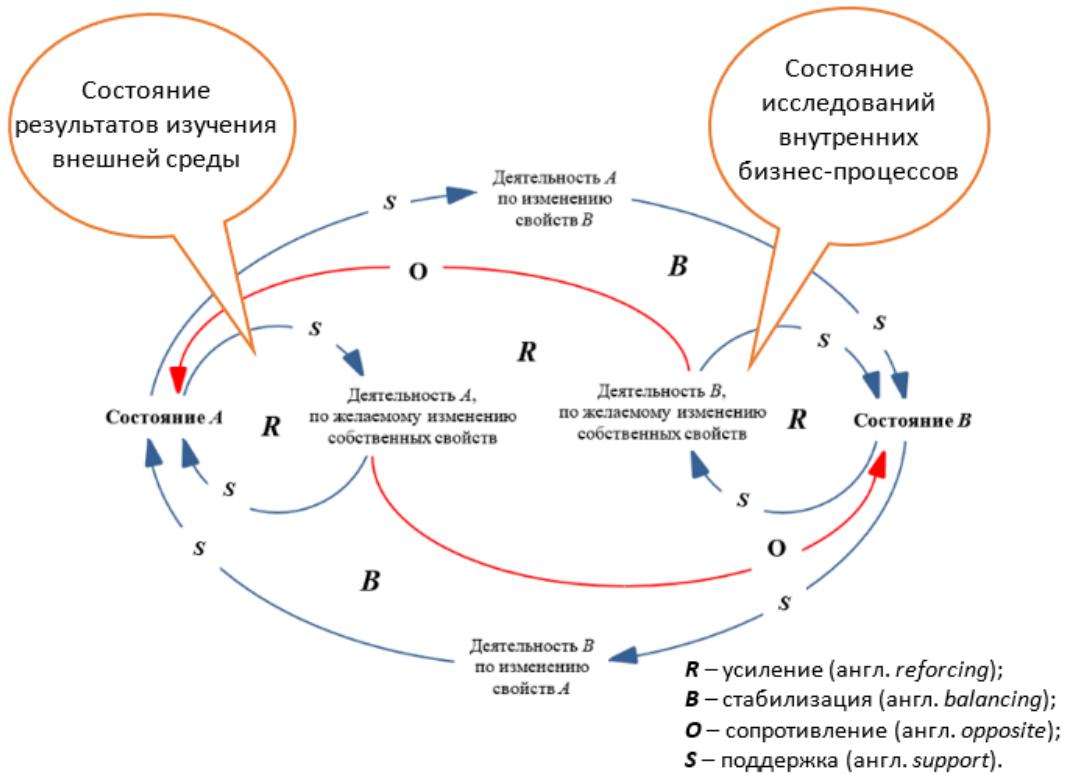


Рисунок 5.10 – Архетип «Случайные противники»

Этому архетипу может быть поставлена в соответствие контурная модель, представленная на рисунке 5.11, основу которой составляет точка зрения на систему управления проектом как на многосвязную систему.

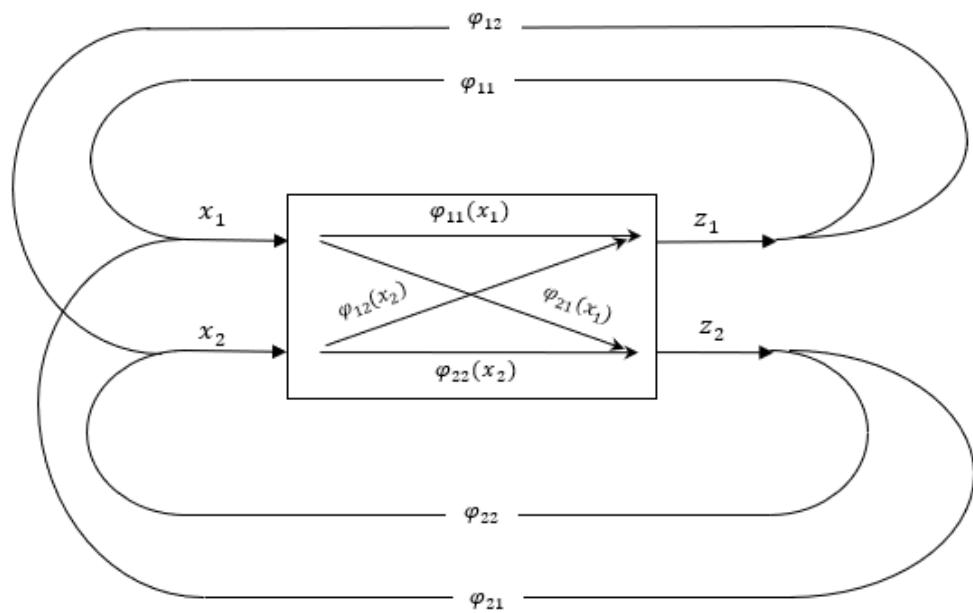


Рисунок 5.11 – Контурная модель, соответствующая системному архетипу «Случайные противники»

В разделах 5.1 и 5.2 на примере задачи определения бюджета проекта рассмотрен подход к описанию проекта как многосвязного объекта управления. Формирование прямых и перекрестных связей проекта  $\varphi_{ij}$  (косвенных характеристик системы управления проектом) осуществляется по критерию удовлетворенности потребителей и исполнителей на основе архивных данных о ранее реализованных проектах и субъективных оценок экспертов. На стр. 184 представлены сведения о входных и выходных характеристиках проекта, а также о функциональных связях между ними. Рассмотрим в случае системного архетипа «Случайные противники».

Характеристики  $y_j$  ( $j = 1, 2$ ) и параметры  $x_i$  ( $i = 1, 2$ ) связаны функциональными зависимостями  $y_j = \Phi_j(x_1, x_2)$ .

$$\Phi_1(x_1, x_2) = \gamma_1 \varphi_{11}(x_1) + \gamma_2 \varphi_{21}(x_2), \quad (5.20)$$

$$\Phi_2(x_1, x_2) = \varphi_{12}(x_1) * \varphi_{22}(x_2), \quad (5.21)$$

где  $\varphi_{ij}(x_i)$ , ( $i, j = 1, 2$ ) – строгие функциональные зависимости. При  $i=j$  они характеризуют прямые связи между входами и выходами объекта; при  $i \neq j$  – перекрестные;

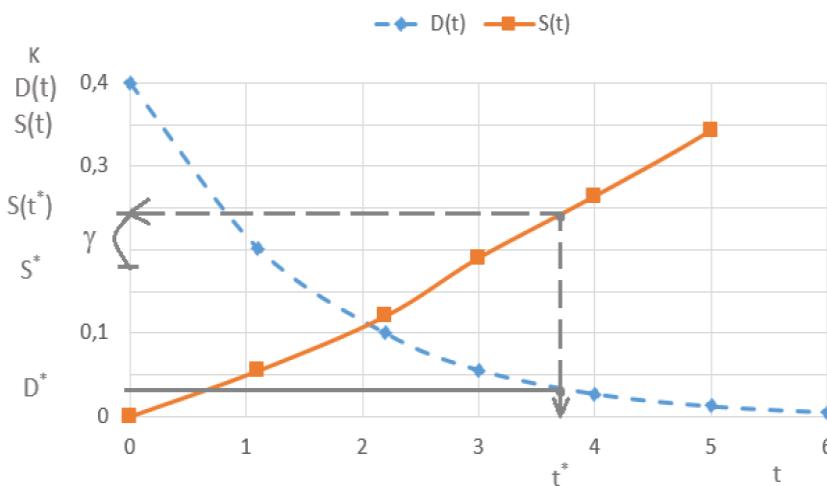
$\gamma_1, \gamma_2$  – коэффициенты, характеризующие значимость параметров  $x_1$  и  $x_2$ ;

$\Phi_{ij}$  – действия, связанные с мониторингом свойств промежуточных результатов проекта и изменением среды реализации проекта ( $i, j = 1, 2$ ).

В рамках сверток (5.20) и (5.21) определяются границы, в которых может изменяться бюджет проекта при наличии ограничений на длительность реализации проекта.

На рисунке 5.12 показан пример, иллюстрирующий ситуацию несбалансированности проекта по показателю бюджета в зависимости от ограничений на допустимое число невыявленных латентных дефектов.

Данный пример, описывает ситуацию, возникшую в следствии несбалансированности нефункциональных требований и объема выделяемых ресурсов, что приводит к превышению ограничений на бюджет и время реализации проекта.



$S^*$  - ограничение на бюджет проекта;  
 $D^*$  - ограничение на допустимое число невыявленных латентных дефектов.

Рисунок 5.12 – Несбалансированность проекта по показателю бюджета

В терминологии системных архетипов манипулирование параметрами проекта на предпроектной стадии может рассматриваться как поиск точки «приложения рычага», с тем чтобы предусмотреть динамику проблемной ситуации и своевременно предпринять меры по снижению негативного влияния проблемных ситуаций на результаты проекта.

Различные проблемные ситуации, возникающие в ходе проекта реализации ИС, обусловлены следующими факторами.

- 1) Многомерностью проекта как объекта управления, включая:
  - неопределенность требований, в особенности в начале жизненного цикла проекта;
  - ограниченность ресурсов (материальных, трудовых, временных, интеллектуальных, других), доступных для реализации проекта;
  - множественность ошибок и дефектов, допускаемых при решении взаимосвязанной совокупности задач проекта;
  - множественность, часто противоречивых, точек зрения ключевых правообладателей на требования к конечным результатам и ходу проекта.
- 2) Неоднородностью акторов, участвующих в том или ином качестве в реализации проекта. Проблемные ситуации, наблюдаемые в рамках разных точек зрения на проект, являются тенью несоответствия внутреннего устройства проекта условиям, в которых он реализуется.

3) Субъектоцентрической природой проектов, недостаточным развитием теоретических основ, методического обеспечения, инструментальных средств предотвращения возникновения субъективных ошибок различной природы, выявлением на стадиях жизненного цикла проекта латентных дефектов, обусловленных этими ошибками.

Организация предпроектного обследования предопределяет результаты проекта. Ошибки, допущенные на предпроектной стадии ИС, имеют тяжелые негативные последствия [130, 131, 200]. В силу этого можно утверждать, что качество управления ресурсами (бюджетом и длительностью проведения обследования) на этой стадии проекта является критическим фактором, предопределяющим дальнейший ход проекта.

Формальной характеристикой организации проекта являются структуры зависимостей, связывающих ресурсы проекта с результатами работ, выполняемых в ходе реализации проекта. Состояние стадии проекта определяется значениями параметров этих зависимостей.

Особенностями проведения предпроектного обследования, результатом которого является спецификация внешнего облика ИС (техническое задание на разработку ИС), являются:

1) Необходимость участия в проведении обследования не только исполнителей проекта, но и представителей организации-заказчика (подчеркивается, например, в отчетах *Standish Group* [143-146]). Субъекты, привлекаемые для проведения обследования могут по-разному влиять на его ход в силу их различного отношения к проекту. Это обусловлено тем, что они имеют различные представления о ценностях, различную квалификацию в предметной области (понимание существа бизнес-процессов, для построения моделей которых они привлекаются разработчиками), их различного отношения к изменению ролей сотрудников организации, обусловленного появлением ИС, и другими причинами.

2) Ограниченные возможности привлечения исполнителем квалифицированных специалистов, обладающих требуемой квалификацией в части формирования внешнего облика ожидаемого результата. Достаточно часто

для выполнения обследования приходится привлекать сотрудников организации-исполнителя, достаточно компетентных в вопросах проектирования и реализации ИС, но не обладающих достаточными знаниями и опытом в области выявления коренных информационных потребностей Заказчика и оформления технического задания.

С учетом указанных обстоятельств одним из системных архетипов, который можно поставить в соответствие стадии разработки технического задания, является «Предел роста» (содержание системных архетипов подробно рассмотрено в разделе 2.5, графическая модель представлена на рисунке 2.12), в котором контур положительной обратной связи (контура развития) определяется бюджетом и длительностью времени реализации проекта, а контур отрицательной обратной связи (торможение) соответствуют факторам, препятствующим получению желаемого результата (к числу которых относятся недостаточная компетентность привлекаемых специалистов, либо их незаинтересованность в реализации проекта). Подробно содержание архетипа рассмотрено в разделе 2.5, графическая модель представлена на рисунке 2.12.

Модель «треугольник проекта» [144] применима также и к отдельным стадиям жизненного цикла проекта, в том числе к стадии определения охвата проекта, т.е. к этапу формирования технического задания.

В литературе, посвященной программным проектам, подчеркивается итерационный характер реализации проектов [30, 55, 67, 85, 100, 115]. Эта особенность программных проектов нашла отражение в известных моделях жизненного цикла. Для этапа формирования охвата проекта итерационный характер выражается в пошаговом согласовании в результате дискурса между заказчиком и руководителем проекта ресурсов этапа и обязательств по потребительским свойствам ИС (что и определяет охват проекта). Итерационный характер разработки подчеркивается наличием контуров обратной связи.

Графическая модель, соответствующая стадии охвата проекта представлена на рисунке 5.13.

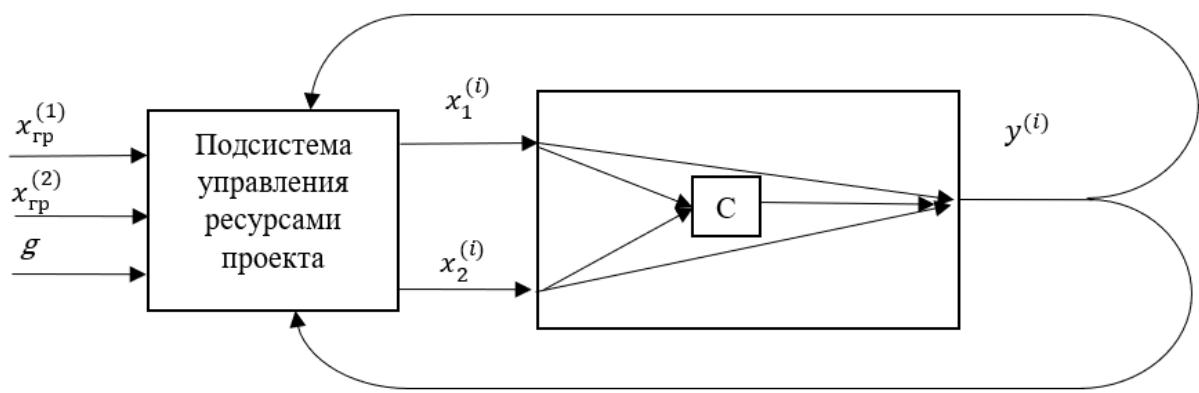


Рисунок 5.13 – Контурная модель, соответствующая стадии охвата проекта

На рис. 5.13 обозначено:

$x_1^{(\text{grp})}$  - верхнее ограничение на бюджет этапа;

$x_2^{(\text{grp})}$  - верхнее ограничение на длительность реализации этапа;

$x_1^{(i)}$  - величина бюджета на  $i$  итерации согласования параметров и ожидаемых результатов этапа между заказчиком и исполнителем;

$x_2^{(i)}$  - длительность реализации этапа на  $i$  итерации согласования параметров и ожидаемых результатов этапа;

$y$  - охват проекта;

$g$  - пожелания к результатам проекта.

Входными параметрами проекта как многосвязного управляемого объекта являются бюджет и длительность реализации, а также нечетко формулируемые пожелания заказчика  $g$  к результатам стадии проекта.

Термином «пожелание» подчеркивается, что на стадии определения охвата проекта формулировки носят достаточно размытый характер, т.е. трудно сформулировать критерий степени реализации пожелания. Примером  $g$  может выступать, например, «...обеспечить наибольший охват потребностей заказчика в условиях ограничений на выделяемые ресурсы...».

Выходными результатами на стадии определения охвата проекта является спецификация внешнего облика ИС. Результаты этапа формирования охвата проекта зависят не только от объемов ресурсов (ограниченные бюджет и

допустимая длительность получения документально зафиксированного охвата проекта), но и от ограничений на квалификацию привлекаемых специалистов и их различную заинтересованность в исходе проекта. Проявление проблемной ситуации проявляется в виде характера изменчивости параметров и результатов проекта.

Как уже отмечалось ранее, организация проекта характеризуется правилами преобразования ресурсов проекта в результат. Формально правила преобразования представляются математическими соотношениями, преобразующими значения входных параметров многосвязного объекта управления в выходные параметры.

Основу построения контурной модели (рисунок 5.13) составляют следующие соотношения между входными и выходными параметрами:

$$y_{i+1} = Ax_1^{(i+1)} + Bx_2^{(i+1)} - Cx_1^{(i+1)}x_2^{(i+1)} \quad (5.22)$$

Закон управления:

$$x_1^{(i+1)} = x_1^{(i)} + k(y_i - y_{i-1}), \quad i = \overrightarrow{0; n} \quad (5.23)$$

$$x_2^{(i+1)} = \varphi(x_1^{(i+1)}) \quad (5.24)$$

Параметры  $A$  и  $k$  характеризуют влияние объема финансовых средств на обязательства, принимаемые исполнителем;

Параметр  $B$  характеризует влияние длительности реализации этапа проекта на обязательства, принимаемые исполнителем;

Параметр  $C$  характеризует компетентность субъектов, привлекаемых к реализации этапа, а также их заинтересованность в результатах проекта.

Условием останова итерационного процесса являются либо выполнение условия  $|y_{i+1} - y_i| \leq \varepsilon$ , что характеризует расхождение в принимаемых обязательствах на  $i$  и  $(i+1)$  итерациях. Содержательно это означает, что дальнейшие изменения управляющих параметров не позволяет значимо изменить соотношение между предлагаемым уровнем охвата (представленным посредством  $y_{i+1}$ ) и  $g$ .

Из литературных источников известно, что в рамках модели «Треугольник проекта» параметры «Бюджет» и «Длительность реализации» не являются

независимыми. В [81] приводятся сведения о наличии обратных зависимостей между стоимостью и длительностью реализации проекта (чем меньше времени выделяется на реализацию проекта, тем больше должен быть его бюджет). В рамках настоящих исследований работы постулируется обратная линейная зависимость вида:

$$x_2^{(i+1)} = a_0 - b_0 * x_1^{(i+1)} \quad (5.25)$$

На рисунке 5.14 представлены графики, характеризующие изменение параметров стадии проекта при разных значениях параметра  $C$ .

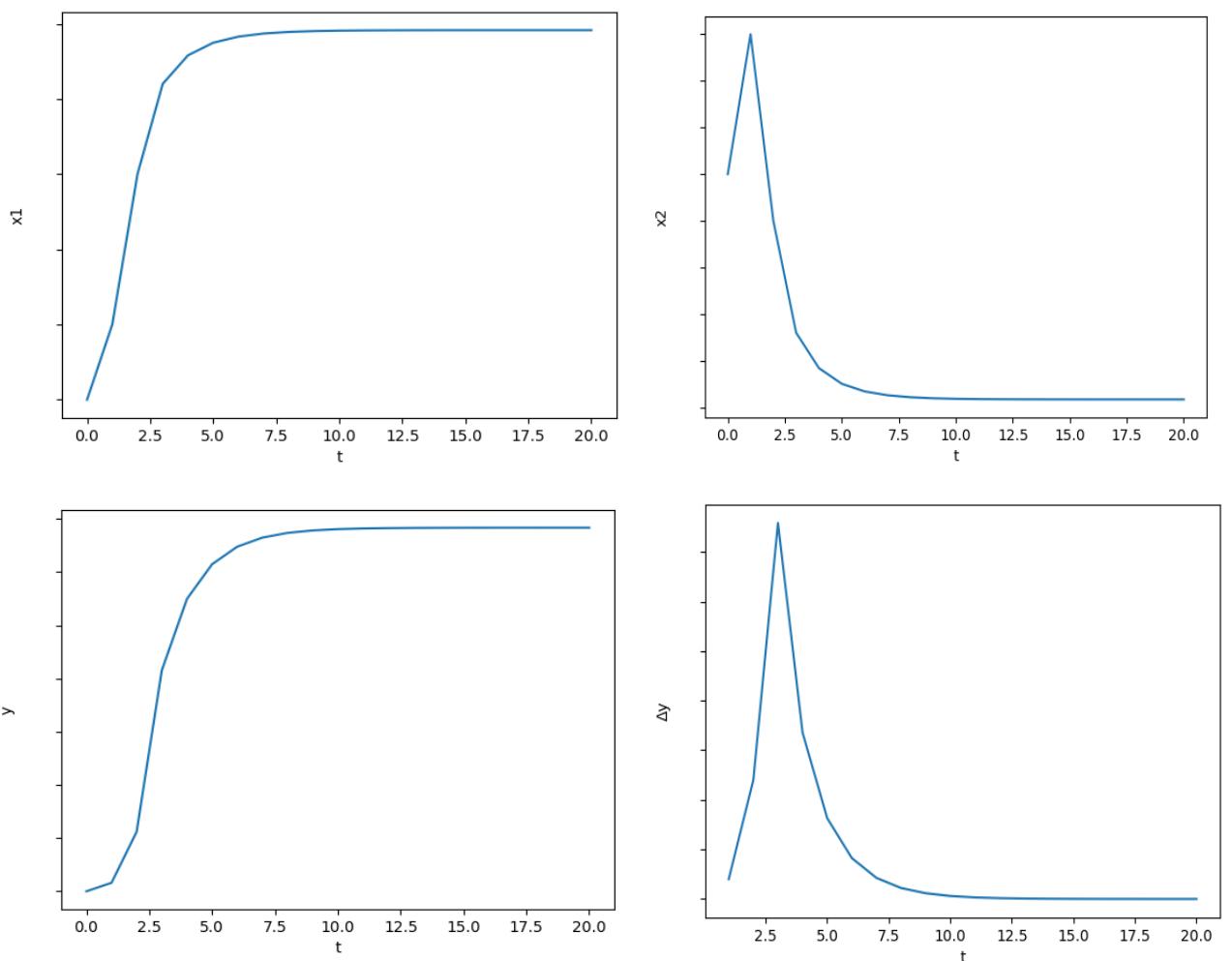


Рисунок 5.14 – Характер изменения параметров проекта  
при разных значениях параметра  $C$

На рисунке 5.15 представлены графики, характеризующие изменение параметров стадии проекта при  $A=2; B=8; C=4; a_0=5; b_0=1; K=1$ .

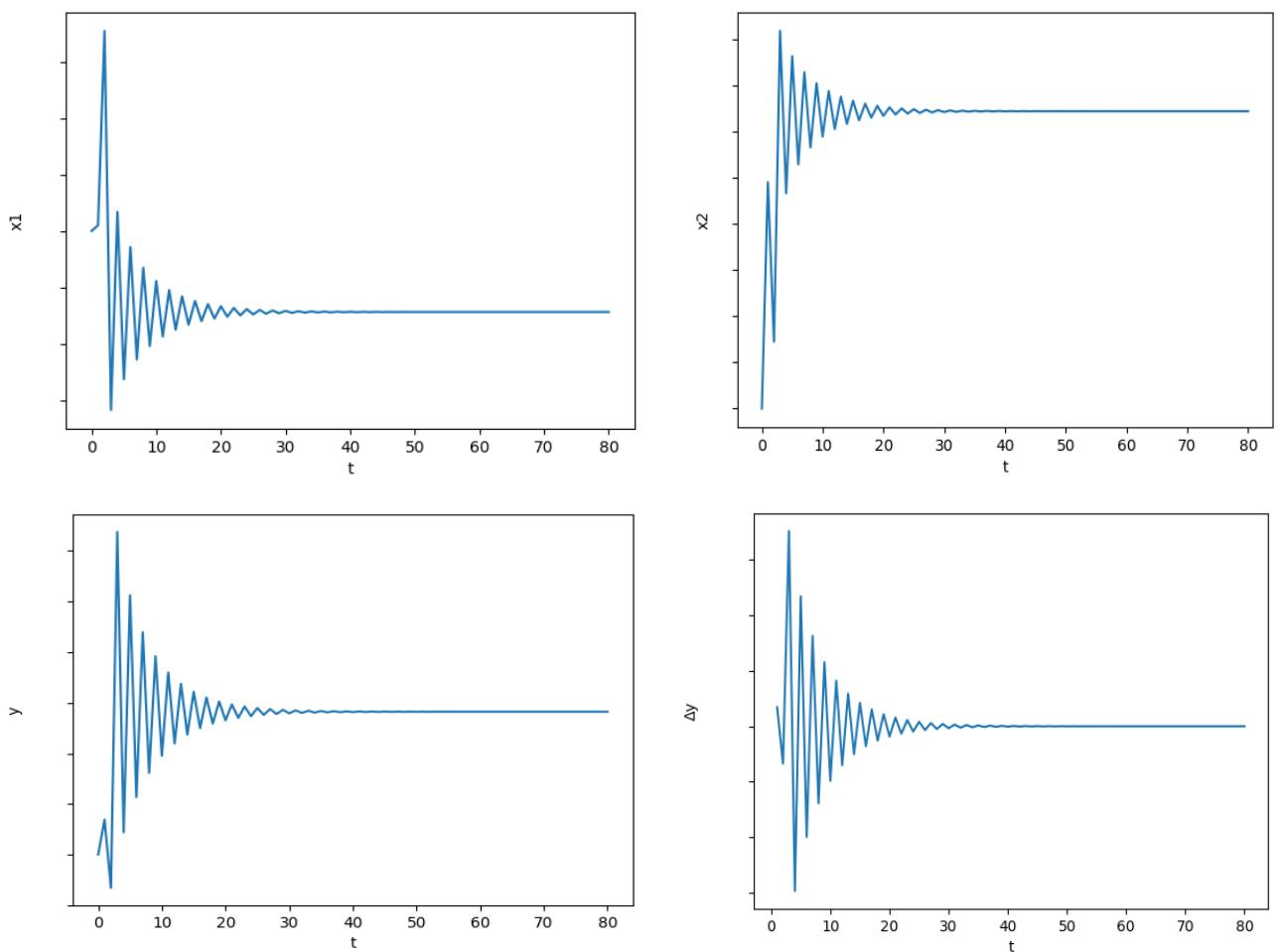


Рисунок 5.15 – Характер изменения параметров проекта

*Пример.* Распределение ресурсов при различиях в требованиях представителей заказчика.

Допустим, в результатах проекта заинтересованы два ключевых правообладателя, у которых представления о ценностях и требования к потребительским свойствам ИС различаются. Реализацией проекта занимается одна команда, причем выделить независимые группы, каждая из которых будет заниматься реализацией требований разных правообладателей, не представляется возможным: члены команды вынуждены параллельно участвовать в разработках в интересах каждого из правообладателей. При этом некоторые виды работ над разными частями проекта могут совпадать. Бюджет проекта и длительность реализации ограничены. Отношение членов команды к задачам, связанным с работами над задачами, соотносимыми с каждым из правообладателей, различно, что выражается в разной степени влияния денег и времени на конечные результаты.

Требуется, распределить ресурсы проекта (бюджет; время) таким образом, чтобы обеспечить получение результата, наиболее близкого к представлениям каждого из правообладателей.

Описанной ситуации можно поставить в соответствие системный архетип «Трагедия общих ресурсов» (содержание системных архетипов подробно рассмотрено в разделе 2.5, графическая модель представлена на рисунке 2.17).

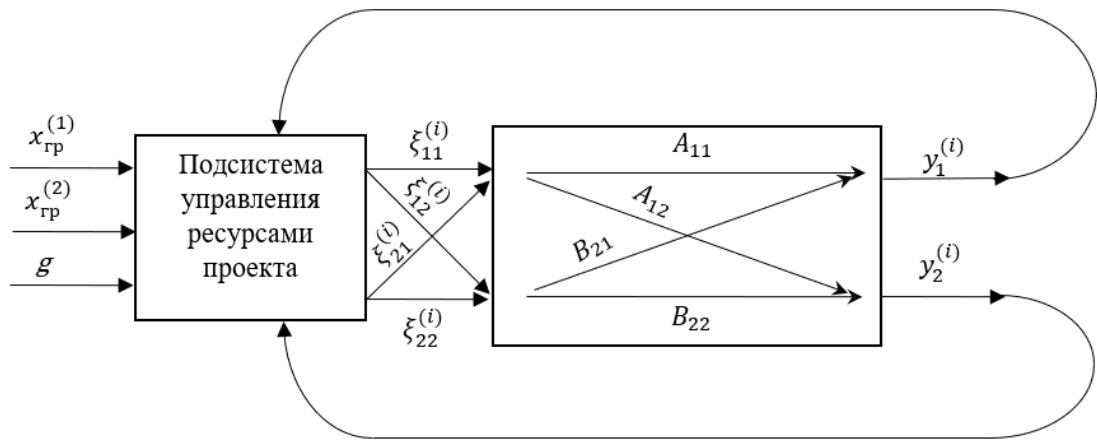


Рисунок 5.16 – Контурная модель, соответствующая условиям различия представлений о качестве разных правообладателей.

Основу построения контурной модели (рисунок 5.16) составляют следующие соотношения между входными и выходными параметрами, характеризующие влияние устройства проекта на выходные результаты  $y_i$ , связанные с каждым из правообладателей:

$$y_1^{(i)} = A_{1,1}\xi_{1,1}^{(i)} + B_{2,1}\xi_{2,1}^{(i)} \quad (5.26)$$

$$y_2^{(i)} = A_{1,2}\xi_{1,2}^{(i)} + B_{2,2}\xi_{2,2}^{(i)} \quad (5.27)$$

Закон управления имеет вид:

$$\xi_{1,1}^{(i+1)} = \xi_{1,1}^{(i)} + D_1 (y_1^{(i)} - y_1^{(i-1)}) - R_{1,1}(y_2^{(i)} - y_2^{(i-1)}) \quad (5.28)$$

$$\xi_{1,2}^{(i+1)} = \xi_{1,2}^{(i)} + D_2 (y_2^{(i)} - y_2^{(i-1)}) - R_{1,2}(y_1^{(i)} - y_1^{(i-1)}) \quad (5.29)$$

$$\xi_{2,1}^{(i+1)} = \xi_{2,1}^{(i)} + E_1 (y_1^{(i)} - y_1^{(i-1)}) - R_{2,1}(y_2^{(i)} - y_2^{(i-1)}) \quad (5.30)$$

$$\xi_{2,2}^{(i+1)} = \xi_{2,2}^{(i)} + E_2 (y_2^{(i)} - y_2^{(i-1)}) - R_{2,2}(y_1^{(i)} - y_1^{(i-1)}) \quad (5.31)$$

Параметры  $x_1, x_2$  означают бюджет и длительность работы над проектом. На эти параметры наложены ограничения  $x_1^{(\text{grp})}$  и  $x_2^{(\text{grp})}$  соответственно.

Начальными условиями являются:  $x_1^{(0)} = x_2^{(0)} = 0; y_1^{(0)} = y_2^{(0)} = 0$ .

Параметры  $\xi_{i,j}$  характеризуют выделение  $i$ -го ресурса на решение задач, соответствующих  $j$ -му правообладателю ( $i, j = 1, 2$ ).

Параметры  $A_{11}, A_{12}$  характеризуют влияние объема денежных средств, выделяемых на выполнение работ в интересах первого и второго правообладателей, на качество результатов  $y_j$ .

Параметры  $B_{21}, B_{22}$  характеризуют влияние времени, выделяемого на реализацию каждой из частей проекта.

Параметры  $D_l$ , ( $l = 1, 2$ ) характеризуют влияние изменения значений  $y_j^{(k)}$  на  $k$ -м шаге, по сравнению со значением на предыдущем шаге, выделение финансовых средств на решение задач, соответствующих  $j$  – му правообладателю.

Соответственно  $E_l$ , ( $l = 1, 2$ ) характеризуют влияние изменения значений  $y_j^{(k)}$  на  $k$ -м шаге, по сравнению со значением на предыдущем шаге, на выделение времени на решение задач, соответствующих  $j$  – му правообладателю.

Параметры  $R_{i,j}$  характеризуют борьбу  $j$ -го правообладателя за  $i$ -е ресурсы.

Перекрестные связи означают, что некоторые результаты, полученные в ходе решения задач, соответствующих разным правообладателям, обладают свойством тиражируемости и могут использоваться при выполнении работ в интересах разных правообладателей.

На рисунке 5.17 представлены графики, характеризующие изменение параметров стадии проекта при разных значениях параметров  $D_1; E_1; E_2; R_{11}; R_{12}; R_{21}; R_{22}$ .

200

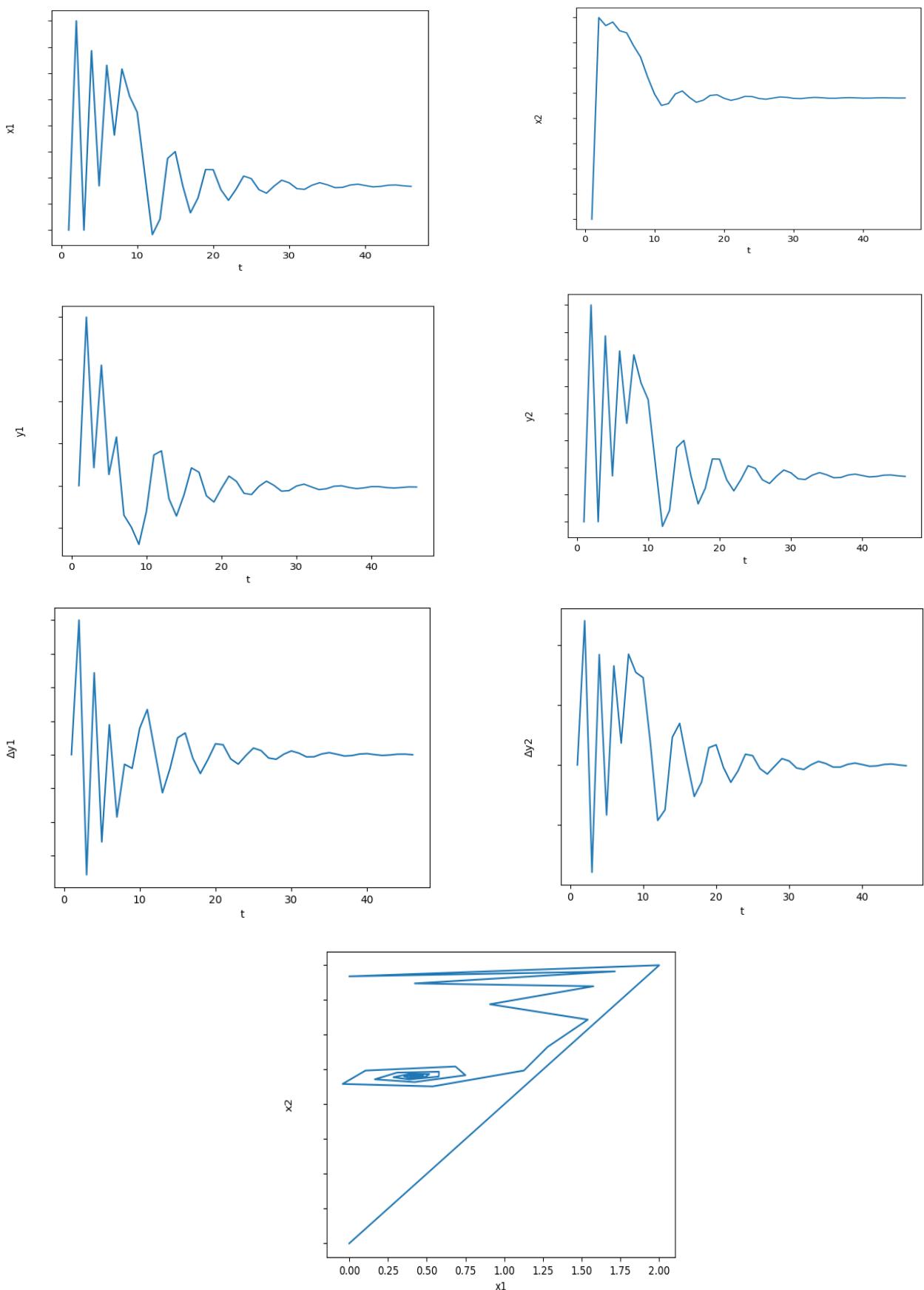


Рисунок 5.17 – Характер изменения параметров объекта управления  
 $(A_{11}=0.1, B_{12}=2, A_{22}=0.3, B_{21}=0.1, C_1=0.2, C_2=0.1; D_1=0.2, D_2=0.3;$   
 $E_1=0.5, E_2=0.1; R_{11}=0.5, R_{12}=4, R_{21}=0.2, R_{22}=0.1)$

При управлении проектами ИС одновременно возникает множество проблемных ситуаций, как в рамках одной точки зрения на проект, так и в рамках разных точек зрения. С целью рационального распределения усилий по формированию подходов и мер к урегулированию проблемных ситуаций и предупреждению их возникновения в будущем, необходимо изучение влияния внутреннего устройства проекта на характер проявления проблемных ситуаций.

Результаты демонстрационных вычислительных экспериментов показывают возможность прогнозировать характер проявления проблемных ситуаций в зависимости от организации проекта. Подход, ориентированный на установление связи внутреннего устройства проекта и характера проявления проблемных ситуаций, позволяет заранее оценить ограничения различных подходов к урегулированию проблемных ситуаций, продумать и подготовить систему мер по парированию негативных последствий проблемных ситуаций.

Предлагаемый подход позволяет повысить формализацию решения задач в рамках подхода, именуемого «барьерным мышлением» [200, 201].

Приведенные результаты соответствуют достаточно простым функциональным зависимостям, связывающим характеристики качества с характеристиками бюджета и длительности реализации проекта. Но это не ограничивает общности подхода, поскольку параметры моделей могут быть композитными, т.е. в свою очередь быть функциями от других параметров. Основу построения таких функций может составить, например, системная модель, известная как *Fishbone*.

Выполнение численного моделирования, ориентированного на получение ситуационной картины этапов проекта на стадии инициализации проекта, позволяет снизить неопределенность, свойственную проекту и за счет этого повысить эффективность и результативность реализации как отдельных этапов проекта, так и проекта в целом.

## 5.5 Оценка согласованности мнений неоднородных акторов на основе системных архетипов

В основе проблемных ситуаций, с которыми приходится сталкиваться при управлении проектами ИС, лежат одни и те же ключевые организационные факторы. Например, организационные ошибки, допущенные ключевыми правообладателями исходя из личных интересов при распределении полномочий и зон ответственности менеджеров, могут послужить причиной ошибок при планировании проекта, выстраивании отношений с представителями заказчика и т.д. Поэтому представляет интерес исследования схожести проблемных ситуаций, имеющих место при реализации проектов на основе совокупного учета персонального понимания содержания проблемных ситуаций неоднородными акторами.

К числу базовых понятий теории интерсубъективного управления [35-39]. относится «конвенционализм». Под конвенциальностью понимается непротиворечивая теория, являющаяся результатом соглашения между неоднородными акторами, формируемая на основе их персонифицированных знаний в результате дискурса. Взаимный обмен неоднородными акторами персонифицированными знаниями создает предпосылки для лучшего понимания природы проблемной ситуации, что, в свою очередь, влияет на обоснованность оценки последствий различных действий по её урегулированию.

В качестве инструмента оценки согласованности мнений неоднородных акторов на предпроектной стадии используются системные архетипы. Отличительной особенностью системных архетипов как основы описания проблемных ситуаций является их ограниченное число. Это позволяет получить сопоставимое описание проблемных ситуаций при разном видении неоднородными акторами. Иными словами, системные архетипы способствуют формированию коммуникационного базиса между неоднородными акторами, задействованными в урегулировании проблемной ситуации.

Разработка формальной схемы создает предпосылки для мониторинга хода процесса выработки признаваемого всеми заинтересованными сторонами видения проблемной ситуации.

Основу предлагаемого подхода составляет следующие:

- содержание проблемной ситуации определяется видением вовлеченных в ее урегулирование неоднородных акторов;
- системные архетипы являются унифицированными качественными моделями типовых проблемных ситуаций, имеющих место в сложных системах разной природы, каждую типовую проблемную ситуацию можно соотнести с определенным системным архетипом;
- проблемная ситуация формально может быть описана совокупностью системных архетипов;
- вклад той или иной типовой проблемной ситуации определяется рангом системного архетипа;
- видению<sup>14</sup> проблемной ситуации каждым неоднородным актором ставится в соответствие ранжированная совокупность системных архетипов;
- преобразование ранжированных совокупностей системных архетипов к виду числовых характеристик позволяет оценить согласованность видения проблемной ситуации разными акторами.

Основу решения задачи составляет модель «Дом качества» (*HoQ*) [187, 161, 140].

«Крыльцо» представляет собой расположенные в определенном порядке системные архетипы.

Каждой  $i$ -й строке «комнаты» соответствует один из известных архетипов.

Столбцы  $Rij$  представляют мнение  $i$ -го актора о вкладе  $j$ -го архетипа в проблемную ситуацию, т.е. являются описанием исследуемой проблемной ситуации посредством системных архетипов.

---

<sup>14</sup> Понятие «видение» (англ.- vision) понимается в смысле, определенном в стандарте ANSI/IEEE Std 1471-2000, Recommended Practice for Architectural Description of Software-Intensive Systems

Столбцы представляют собой совокупность рангов, присваиваемых актором каждому из системных архетипов. Допускается присвоение разным системным архетипам одинаковых рангов.

«Крыша» содержит значения парных коэффициентов корреляции, вычисленных посредством известного соотношения на основе ранговых коэффициентов корреляции Кенделла [62, 64]:

$$K_{ij} = \frac{P_{ij} - Q_{ij}}{\sqrt{\left[ \frac{1}{2} n(n-1) - P_{ij} \right] \left[ \frac{1}{2} n(n-1) - Q_{ij} \right]}}, \quad (5.32)$$

где  $P_{ij}$  – число совпадений рангов, присваиваемых  $i$ -м и  $j$ -м акторами;

$Q_{ij}$  – число инверсий (несовпадений) рангов, присваиваемых  $i$ -м и  $j$ -м акторами;

$n$  – число системных архетипов (десять).

Формула для перевода ранговых коэффициентов корреляции в парные [53]:

$$r_{ij} = \sin\left(\frac{K_{ij}}{2} * \pi\right) \quad (5.33)$$

Преобразование субъективных оценок неоднородных акторов относительно значимости отдельных типовых проблемных ситуаций к виду квадратной матрицы парных коэффициентов корреляции позволяет задействовать инструменты корреляционного анализа для дальнейших исследований.

Сведение задачи к такому виду обеспечивает возможность оценить значения как множественных, так и парциальных коэффициентов корреляции.

Множественные коэффициенты корреляции в данном контексте выступают в качестве количественных характеристик степени совпадения восприятия отдельным актором некоторой грани проблемной ситуации с совокупным видением остальных акторов.

Парциальные коэффициенты корреляции рассматриваются как степень совпадения видения проблемной ситуации между парами акторов.

Построение матрицы парных корреляций создает основу для исследования взаимовлияния состояний, соответствующих разным видениям, на основе парциальных и множественных коэффициентов корреляции.

*Пример использования предлагаемого подхода*

Рассмотрим три ситуации, в которых первая соответствует слабой коррелированности взглядов акторов, вторая - средней коррелированности, третья - сильной коррелированности.

Схема моделирования в рамках предложенного подхода сводится к реализации следующей последовательности шагов.

1) Расположим десять системных архетипов в некоторую последовательность (содержание системных архетипов представлено в разделе 2.5). Порядок расположения архетипов произвольный, однако в ходе эксперимента не изменяется.

2) Допустим, имеются три актора, и каждый из них присваивает системным архетипам определенный ранг. Значения рангов формировались посредством датчика равномерно распределенных случайных чисел.

3) Для  $i$ -го актора ( $i = \overline{1; 3}$ ) каждому архетипу присваиваем значение  $R_{ij} = \text{int}(10 * \xi_{ij})$ . Значения  $\xi_{ij} \in [a_{ij}^{(k)}, b_{ij}^{(k)}]$  определяются границами, в которых возможно изменение значения  $j$ -го архетипа в  $k$ -й ситуации.

4) Сгенерируем значения, равномерно распределенные на заранее заданных интервалах:

- в случае слабой корреляции ( $k=1$ ) для всех акторов случайные числа лежат в диапазоне  $[0.1; 1]$ ;

- в случае средней корреляции ( $k=2$ ):

для первого - пятого системных архетипов  $\xi_{ij} \in [0.1; 0.5]$ ,

для шестого-десятого системных архетипов  $\xi_{ij} \in [0.6; 1.0]$ .

- в случае сильной корреляции ( $k=3$ ):

для первого - второго системных архетипов  $\xi_{ij} \in [0.1; 0.2]$ ,

для третьего и четвертого системных архетипов  $\xi_{ij} \in [0.3; 0.4]$ ,

для пятого и шестого системных архетипов  $\xi_{ij} \in [0.5; 0.6]$ ,

для седьмого и восьмого системных архетипов  $\xi_{ij} \in [0.7; 0.8]$ ,

для девятого и десятого системных архетипов  $\xi_{ij} \in [0.9; 1.0]$ ,

5) Допустим, до начала обсуждения состояния проекта неоднородные акторы назначали ранги системным архетипам так, как указано в «Комната» дома качества (рисунок 5.18) «Крыша», содержит значения парных коэффициентов корреляции, соответствующих исходному упорядочиванию системных архетипов.

6) После первого обсуждения видений проблемной ситуации посредством дискурса (этому соответствует схема генерации случайных рангов, соответствующая  $k=2$ ) в результате преобразований парных коэффициентов корреляции рассчитаем множественные и парциальные коэффициенты корреляции (результаты представлены на рисунке 5.19).

7) Выполним аналогичные операции при втором дискурсе, этому соответствует схема генерации случайных рангов, соответствующая  $k=3$ , (результаты представлены на рисунке 5.20).

8) Значения множественных и парциальных коэффициентов корреляции представлены в таблице 5.1. На рисунке 5.21 представлены графики, характеризующие изменения значений коэффициентов корреляции.

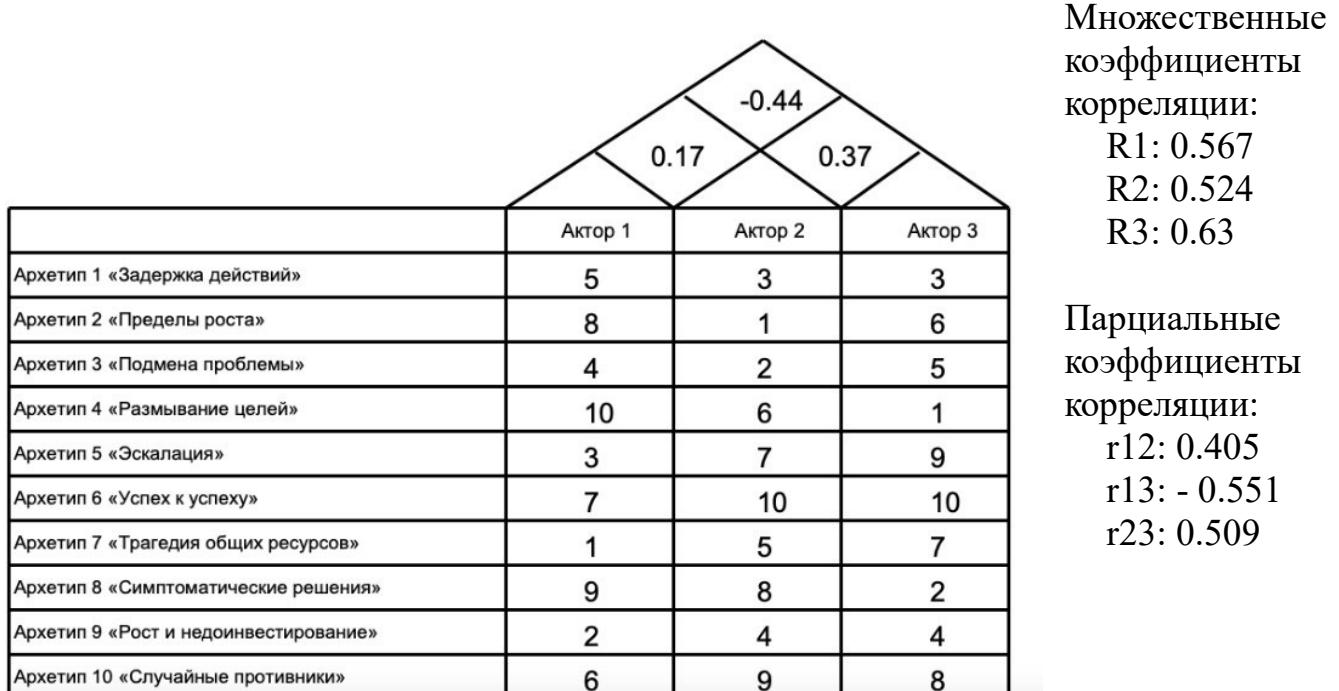


Рисунок 5.18 – Вид «Дома качества» до начала обсуждений

проблемной ситуации



Множественные коэффициенты корреляции:

R1: 0.835  
R2: 0.628  
R3: 0.813

Парциальные коэффициенты корреляции:

r12: 0.335  
r13: 0.711  
r23: 0.132

Рисунок 5.19 – Вид «Дома качества» после первого дискурса



Множественные коэффициенты корреляции:

R1: 0.925  
R2: 0.925  
R3: 0.9

Парциальные коэффициенты корреляции:

r12: 0.608  
r13: 0.4  
r23: 0.4

Рисунок 5.20 – Вид «Дома качества» после второго дискурса

Таблица 5.1 – Значения множественных и парциальных коэффициентов корреляции на разных этапах дискурсов

Номер испытания	Множественные коэффициенты корреляции			Парциальные коэффициенты корреляции		
	R1	R2	R3	r12	r13	r23
k=1	0.567	0.524	0.63	0.405	- 0.551	0.509
k=2	0.835	0.628	0.813	0.335	0.711	0.132
k=3	0.925	0.925	0.9	0.608	0.4	0.4

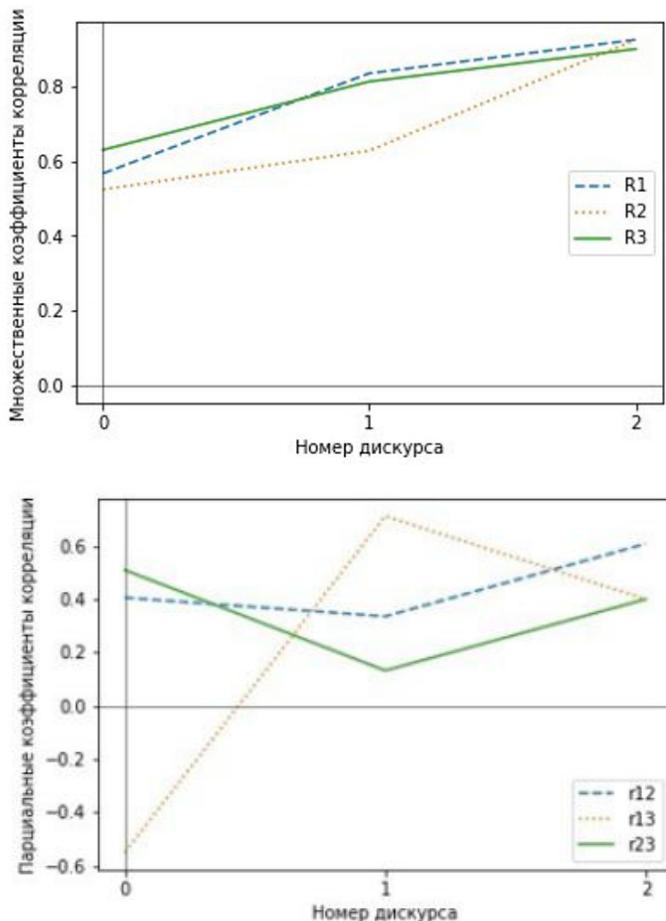


Рисунок 5.21 – Изменение значений коэффициентов корреляции

Конвергенция<sup>15</sup> видений проблемной ситуации формально выражается в том, что на каждом шаге наблюдается сближение значений множественных коэффициентов корреляции. Различие между значениями парциальных коэффициентов корреляции также уменьшается.

Предлагаемый подход к оцениванию степени совпадения точек зрения акторов на основе их субъективного восприятия значимости проблемных ситуаций, позволяет оценивать знак тенденции и скорость сходимости (в случае положительной тенденции), что делает возможным оценить качество организации дискурсов, а также оценивать временные затраты на достижение приемлемого, с точки зрения урегулирования проблемной ситуации, совпадения взглядов правообладателей.

<sup>15</sup> Конвергенция – способ совместного решения коллективом людей в случае, когда всех участников сближают сходные проблемные условия, которые заставляют формировать цель и стратегию урегулирования проблемных ситуаций, согласно [39].

## Выводы по главе 5

1) Предложена модель проекта как многосвязного объекта управления. Входными параметрами модели являются бюджет проекта и длительность реализации проекта. Выходными параметрами являются удовлетворенность заказчика результатами проекта и удовлетворенность исполнителя ходом проекта. Эмпирические функциональные зависимости соответствуют прямым и перекрестным связям между входными и выходными параметрами многосвязного объекта. Особенностью построения эмпирических моделей является использование как фактических (исторических) данных о бюджетах и продолжительности ранее реализованных проектов, так и субъективных экспертных оценок участников проектов (заказчиков и исполнителей).

2) Рассмотрен процесс формирования сбалансированной системы характеристик проекта. На основании полученных зависимостей с учетом реальных ограничений на бюджет и длительность реализации проекта на предпроектной стадии можно оценить предполагаемое качество результатов (выражаемое степенью удовлетворенности) с учетом прежнего опыта взаимодействия исполнителя и заказчика. Это позволяет повысить обоснованность решений о целесообразности реализации проекта силами предполагаемого исполнителя с учетом приоритетности бюджета и длительности реализации проекта для заказчика.

3) Предложен подход к оцениванию одной из основных характеристик проекта – бюджета, опирающийся на эмпирические модели, основу построения которых составляет опыт реализации предыдущих проектов. Принципиальным отличием предлагаемого подхода к оценке бюджета проекта является ориентация не только на трудоемкость разработки, но и на приемлемый как для потребителя, так и для исполнителей уровень удовлетворенности.

4) Разработаны контурные модели проблемных ситуаций, возникающих при реализации проектов ИС, на основе системных архетипов, весовые характеристики которых определяются математико-статистическими методами, это позволяет перейти от качественного описания ситуаций к математическим моделям, позволяющим получить количественные оценки отдельных ситуаций.

5) Предложен подход к оцениванию степени совпадения точек зрения неоднородных акторов на основе их субъективного восприятия значимости проблемных ситуаций, основанный на использовании системных архетипов как основы формализации описания видений неоднородных акторов, модели «Дом качества» и коэффициентов корреляции. Предложена схема формирования метрических характеристик степени согласованности мнений неоднородных акторов.

Основные результаты, представленные в пятой главе, опубликованы в рецензируемых печатных изданиях из перечня, утвержденного ВАК [11, 14, 26] в международных журналах, индексируемых *WoS* [131, 132], в разделах монографии [118] и в других изданиях [26, 28, 119].

## ГЛАВА 6. ПРАКТИЧЕСКОЕ ИСПОЛЬЗОВАНИЕ НАУЧНЫХ РЕЗУЛЬТАТОВ В ЗАДАЧАХ ОБЕСПЕЧЕНИЯ ФУНКЦИОНАЛЬНОЙ НАДЕЖНОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ

Данная глава посвящена решению практических задач обеспечения функциональной надежности ИС на основе полученных в предыдущих разделах теоретических результатов.

Во первом параграфе рассматриваются вопросы повышения функциональной надежности ИС при формировании единого информационного пространства компании.

В втором параграфе представлена методика формальной оценки возможности реализации проекта. В основе методики полученные теоретические результаты, основу оценивания возможности реализации проекта составляет нахождение таких параметров проекта (бюджета и длительности реализации), при которых уровень удовлетворенности заказчика и исполнителя не ниже некоторых заданных значений.

В третьем параграфе представлена методика и программное обеспечение для для оценки возможных последствий отказов функциональных компонент ИС по критериям функциональной надежности. Предложенные решения ориаются на полученные теоретические результаты, основу которых составляет метод структурного моделирования последствий отказов с использованием схем сопряжения и таблиц истинности.

## **6.1 Выявление факторов функциональной надежности информационных систем при формировании единого информационного пространства компании**

В данном разделе решается практическая задача выявления факторов функциональной надежности ИС при формировании единого информационного пространства Инжиниринговой компании НПФ ЭИТЭК.

«Научно-производственная фирма ЭИТЭК» (ООО «НПФ ЭИТЭК») – российская инжиниринговая компания, осуществляющая полный цикл работ по строительству новых, реконструкции и модернизации действующих технологических установок на предприятиях нефтедобывающей, нефтеперерабатывающей и нефтехимической промышленности.

Компания разрабатывает современные инжиниринговые решения для промышленных предприятий. Направления деятельности компании:

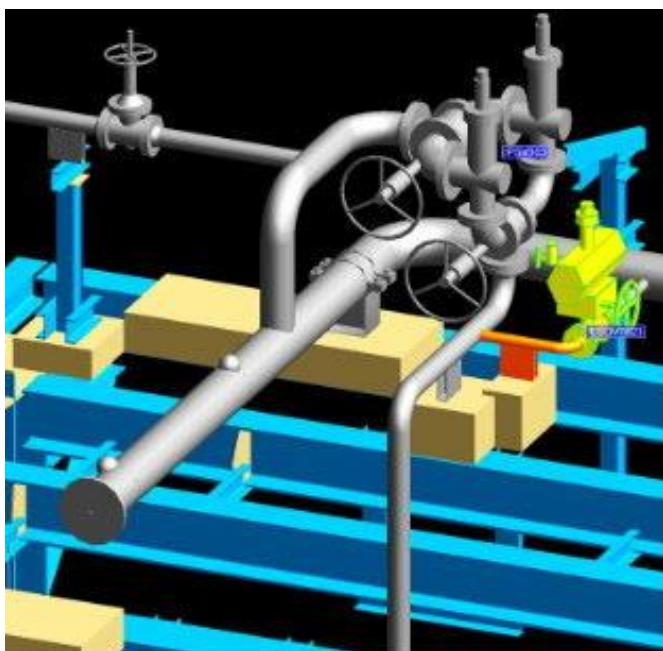
- Информационное моделирование и ИТ-разработки;
- Технологические установки НПЗ;
- Водооборотные системы, системы подготовки и очистки воды;
- Системы АСУТП и КИПиА.

В настоящее время в компании используются разнообразные информационные системы, решающие отдельные локальные задачи. В частности, используются такие системы как:

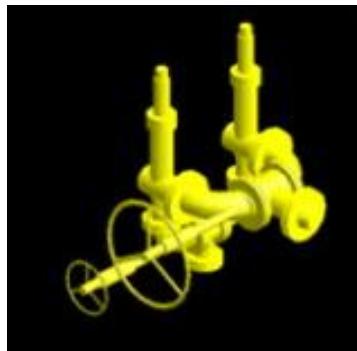
- Aspen HYSYS для технологических расчетов;
- системы автоматизированного проектирования КОМПАС-3D, AutoCad;
- СТАРТ и Гидросистема для прочностных расчетов;
- КОМПАС-Электрик для проектирования электрических сетей и др.

В компании активно работают с технологиями информационного моделирования. Данные технологии предоставляют возможность увидеть реализуемый объект в комплексе, что совместно с высоким качеством проектирования минимизирует ошибки монтажа и отклонений от проекта.

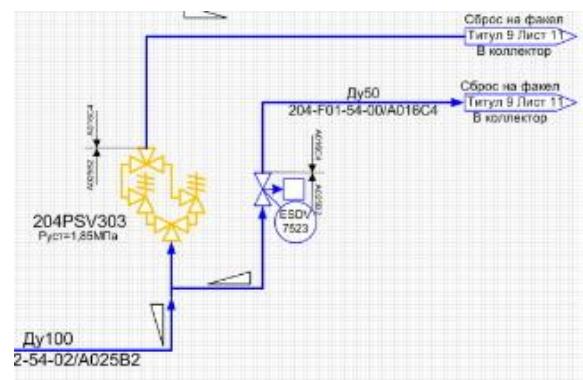
Фрагмент модели показан на рисунке 6.1.



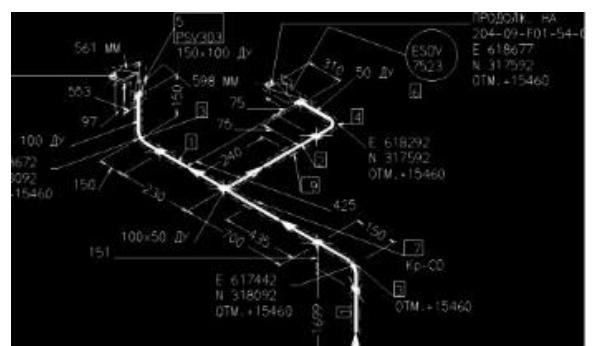
каталог



технологическая схема



изометрия



чертежи



Рисунок 6.1 – Пример информационного моделирования<sup>16</sup>

В последнее время компания инвестировала значительные ресурсы в разработку комплексной инфраструктуры управления знаниями внутри организации, чтобы эффективно использовать технологии и опыт предыдущих проектов с целью формирования единого информационного пространства.

<sup>16</sup> Источник - официальный сайт Инженеринговой компании НПФ ЭИТЭК [www.eitec.ru](http://www.eitec.ru)

Формирование единого информационного пространства посредством бесшовной информационной интеграции локальных ИС связано с определенными проблемами.

Далее представлены результаты анализа ошибок и проблем, с которыми сталкиваются пользователи локальных ИС. На основе реактивного подхода (применен метод анализа коренных причин, *англ. RCA*), выявлены первопричины, способствующие возникновению нежелательных событий в ИС.

Выбор RCA как одного из методов анализа обусловлен его способностью выявлять латентные дефекты в ИС, которые не всегда обнаруживаются при стандартном мониторинге. Применение RCA позволяет систематизировать накопленные данные службы поддержки и использовать их не только для устранения инцидентов, но и для корректировки проектных решений

С целью выполнения анализа ошибок, с которыми встречаются пользователи при работе с ИС, проведен кластерный анализ поступивших запросов в службу поддержки.

Применение кластерного анализа позволило структурировать большой массив разнородных обращений, выявив закономерности в их распределении. Такой подход обеспечивает переход от индивидуального реагирования на инциденты к системному анализу информации, что является необходимым условием при формировании единого информационного пространства.

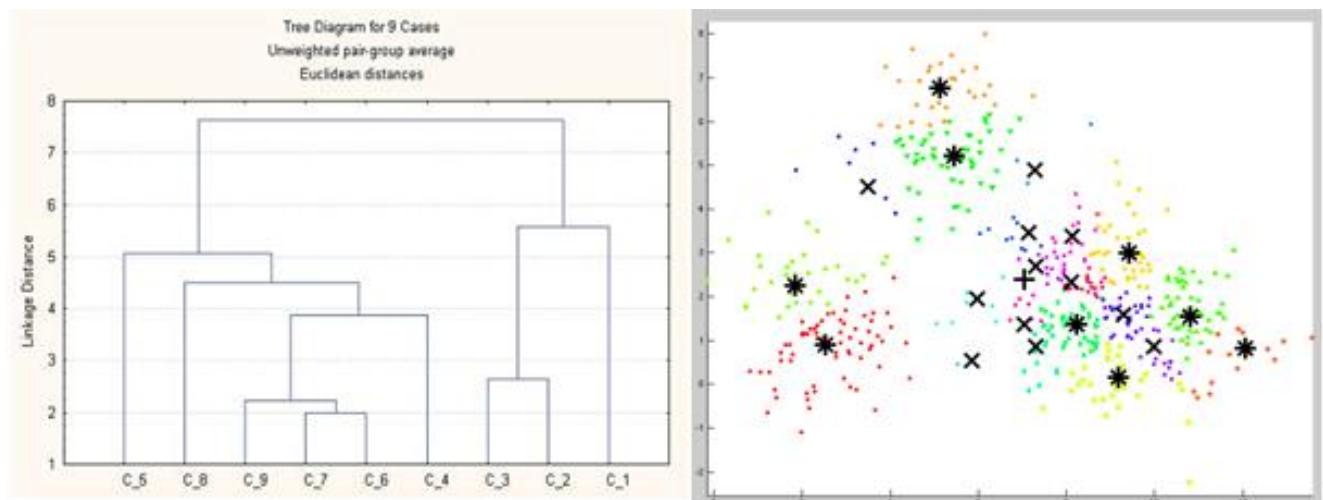
Выделение кластеров происходило на основе зарегистрированных запросов от пользователей в службу поддержки с 2019 года. Основными признаками для кластеризации выделены следующие атрибуты запроса:

- приоритетность запроса;
- время на решение запроса;
- группа запроса;
- текст запроса;
- подкатегория запроса.

Сложность кластеризации состоит в том, что атрибут запроса «текст запроса» не формализован, и каждый пользователь пишет текст произвольно. Для того,

чтобы справиться с этой проблемой при кластеризации использовались ключевые слова и словосочетания такие как: ошибка авторизации, вход в систему, ошибка входа, отчет, неверные данные, отсутствие данных, нет данных, шаблон, сохранение данных и т.д.

В результате выполнения кластеризации выделены 9 основных кластеров (рисунок 6.2). Результат распределения по кластерам представлен в таблице 6.1.



«\*» - центры кластеров, «х» - области неопределенных точек

Рисунок 6.2 – Выявленные кластеры запросов

Таблица 6.1 – Распределение количества запросов по кластерам

Кластер	Количество запросов	% запросов от общей суммы
Не удается авторизоваться в системе	250	9
Не выгружается отчет из модуля	494	19
Модель не добавляется в систему	385	15
Неверное представление данных в отчете	664	25
Неверный шаблон представления данных	20	1
Сбой рассылки отчетов	150	6
Не сохраняются данные в БД	376	14
Не загружаются данные из файла	241	9
Прочие	60	2

Для обоснования распределения ресурсов на обработку запросов, входящих в тот или иной кластер, полученные результаты нуждаются в дальнейшем анализе, а также в ранжировании по приоритетности.

На рисунке 6.3 представлена диаграмма Парето для выделенных кластеров запросов. На диаграмме Парето отображается распределение запросов в порядке убывания их частоты. Линия совокупных значений на дополнительной оси отображает процент от итоговой суммы.

В соответствии с диаграммой Парето, 80% последствий ошибок происходят от 20% причин. Это позволяет распределить усилия для разрешения возникающих проблем и установить основные факторы, на которые необходимо воздействовать с целью преодоления возникающих нежелательных проблемных ситуаций.

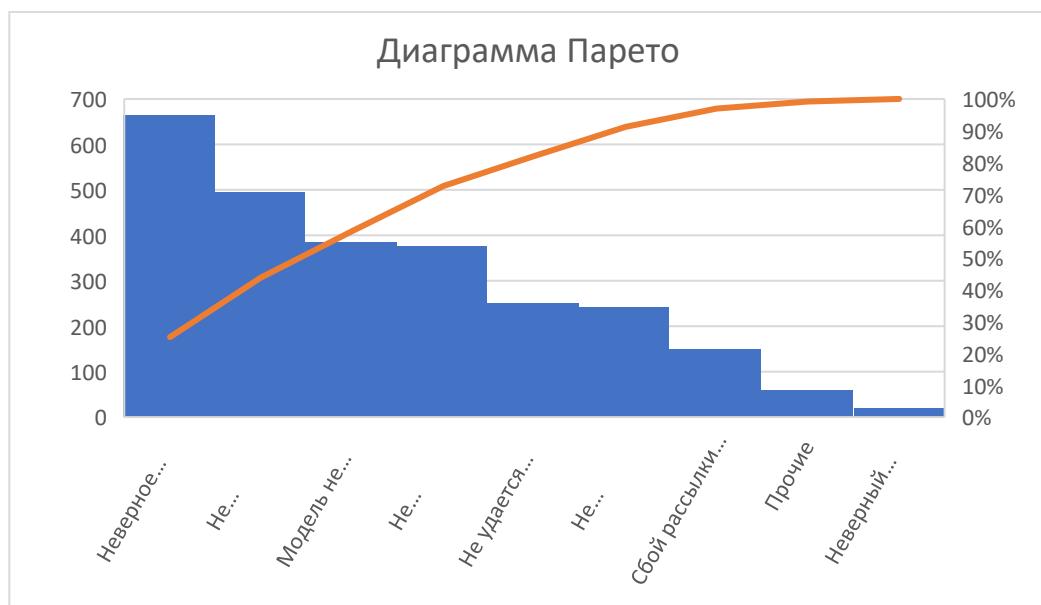


Рисунок 6.3 – Диаграмма Парето

Диаграмма Парето демонстрирует, что устранение ограниченного числа ключевых причин позволяет достичь значительного эффекта в повышении надежности ИС.

На основе выделенных кластеров запросов, построена диаграмма Исикавы (*Fishbone*), представленная на рисунке 6.4. Диаграмма представляет собой визуальный инструмент для выявления первопричин проблем, а также способствует разноспектному комплексному изучению факторов, выявляя первопричины и формируя представление о проблемной ситуации.

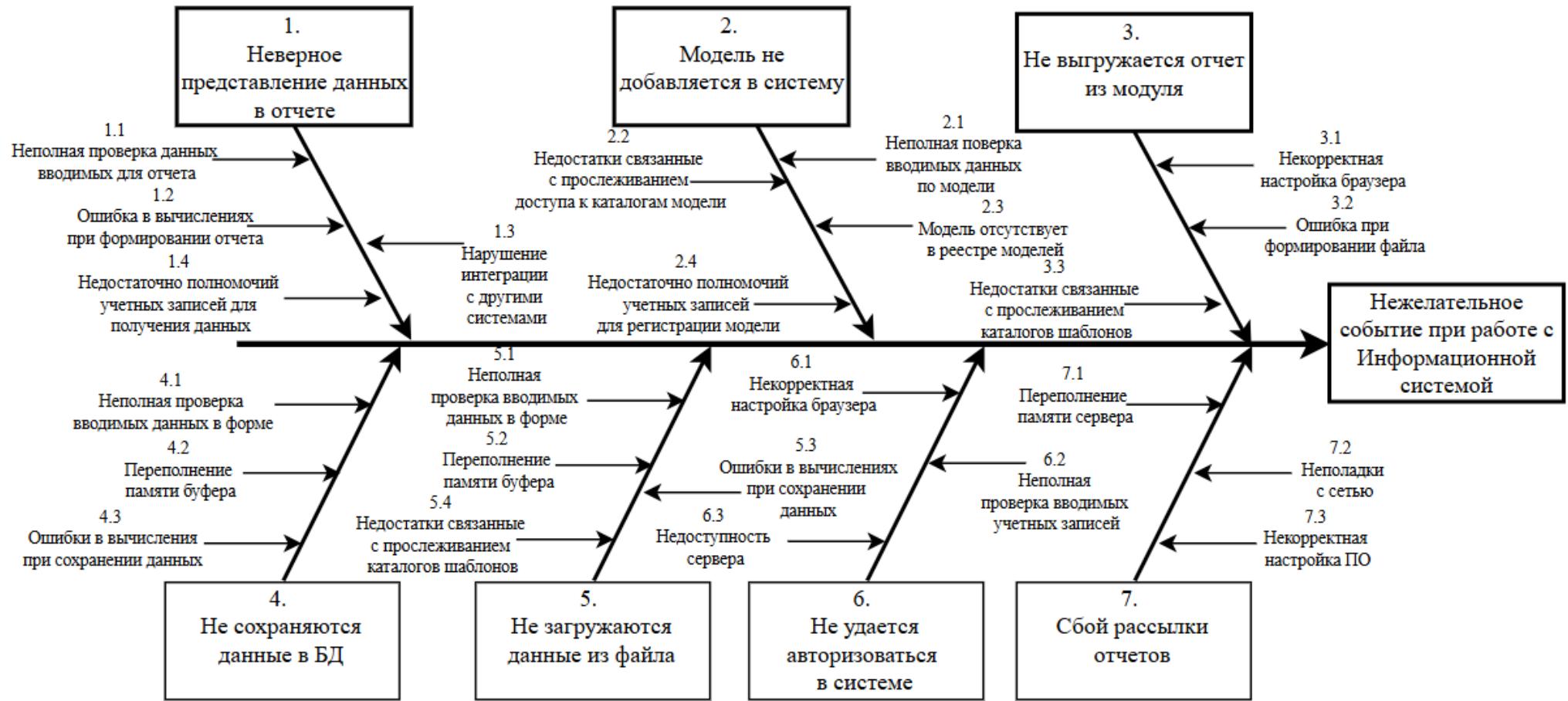


Рисунок 6.4 – Диаграмма Исикавы

Выявленные кластеры запросов пользователей являются следствием некорректной работы ИС, т.е. нежелательным событием в информационной системе. Из построенной диаграммы (рисунок 6.4), можно установить причинно-следственную связь между причинами первого уровня и нежелательными событиями.

Установлено, что одна и та же причина может привести к разным событиям, например, неполная проверка вводимых данных в форме ввода может влиять на возникновение события, приводящего к невозможности сохранить данные в БД, и к событию, ведущему к невозможности загрузить данные из файла, или некорректная настройка браузера может повлечь за собой невозможность выгрузки отчета из системы или к невозможности пройти авторизацию в системе.

В качестве примера детализации на рисунке 6.5 представлена диаграмма Исикавы для кластера «Неверное представление данных в отчете». Остальные диаграммы представлены в приложении Б.

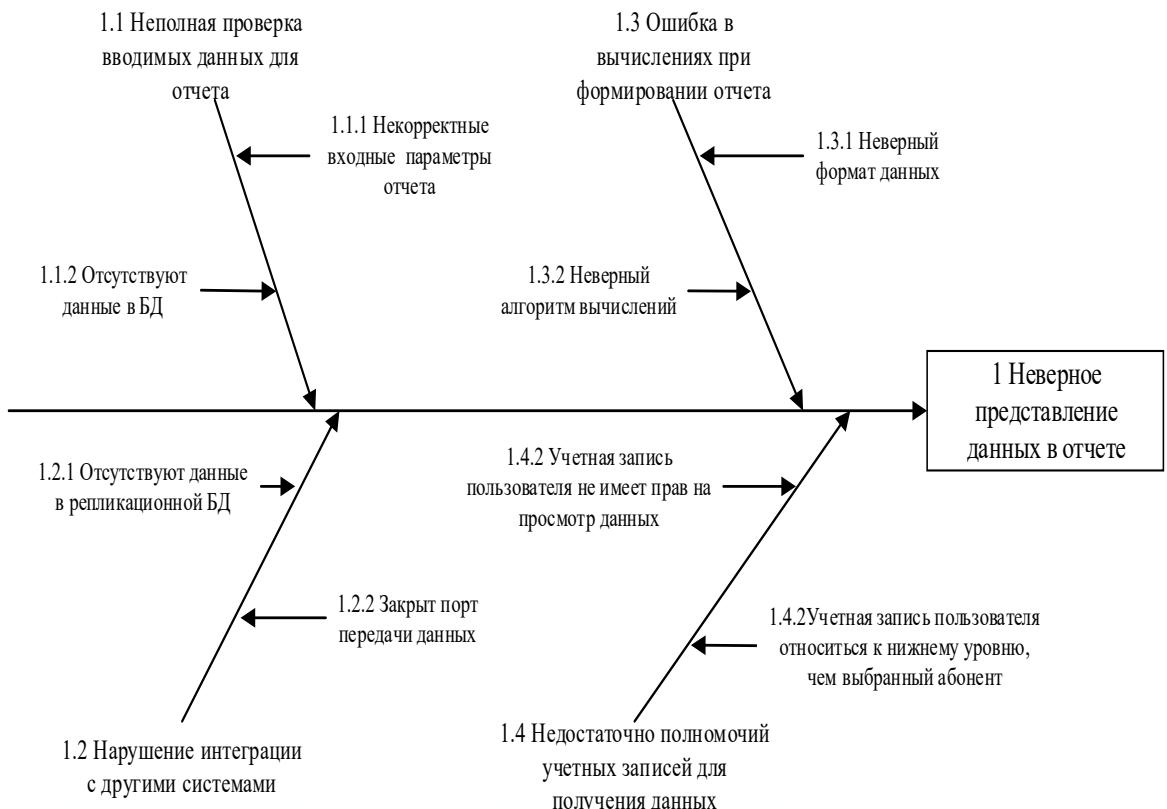


Рисунок 6.5 – Диаграмма Исикавы. Детализация для кластера «Неверное представление данных в отчете»

Пользователь встречается с данной проблемной ситуацией при попытке выгрузить отчет из модуля. Как правило, в такой ситуации отчет формируется в формате excel, но данные в этом отчете неверны или совсем отсутствуют. Причинами этого могут быть:

- неполная проверка вводимых данных для отчета;
- нарушение интеграции с другими системами;
- ошибки в вычислениях при формировании отчета;
- недостаточно полномочий учетных записей для получения данных.

В свою очередь данные причины являются так же следствием при возникновении других причин, которые в совокупности приводят к неверному представлению данных в отчете.

Таким образом, построение диаграмм Исиавы позволяет выявить причинно-следственные связи, на основе реактивного подхода позволяет сосредоточиться на содержании проблемной ситуации, является основой для дискуссии по выявлению коренных причин дефектов.

На следующих этапах анализа функциональной надежности используем метод анализа дерева отказов (*англ. FTA*). Анализ дерева отказов (*FTA*) позволяет наглядно представить логику возникновения причин, приводящих к нежелательному событию (следствию) в локальной ИС.

Применение *FTA* оправдано тем, что он позволяет перейти от констатации факта отказа к формированию логической модели причинно-следственных связей. Построение иерархии причин разного уровня дает возможность выделить критические точки, воздействие на которые наиболее эффективно с точки зрения предотвращения дальнейших отказов.

По своей сути *FTA* реализует реактивный подход, однако результаты такого анализа помогают разработать меры, направленные на снижение вероятности возникновения нежелательных событий, т.е. реализовать проактивный подход.

На рисунке 6.6 показана схема первого уровня дерева отказов, которая представляет причинно-следственную связь возникновения нежелательного события в информационной системе.

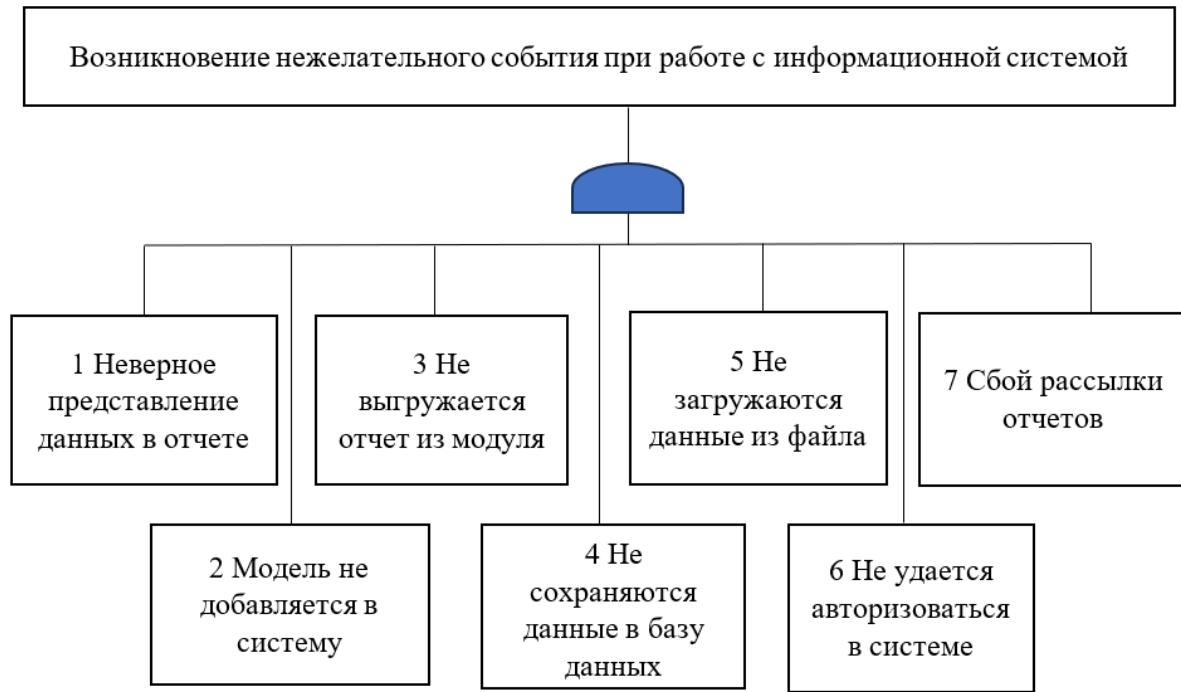


Рисунок 6.6 – FTA. Нежелательное событие при работе с информационной системой

Так как причины являются независимыми друг от друга, то при возникновении хотя бы одной причины происходит возникновение нежелательного события в системе, следовательно, необходимо рассмотреть каждую причину отдельно от других.

В качестве примера, на рисунке 6.7 представлено дерево отказов для причины «Неверное представление данных в отчете». Чтобы выявить причины нижнего уровня необходимо декомпозировать данную диаграмму.

Пример декомпозиции показан на рисунке 6.8. FTA для остальных причин представлены в приложении Б.

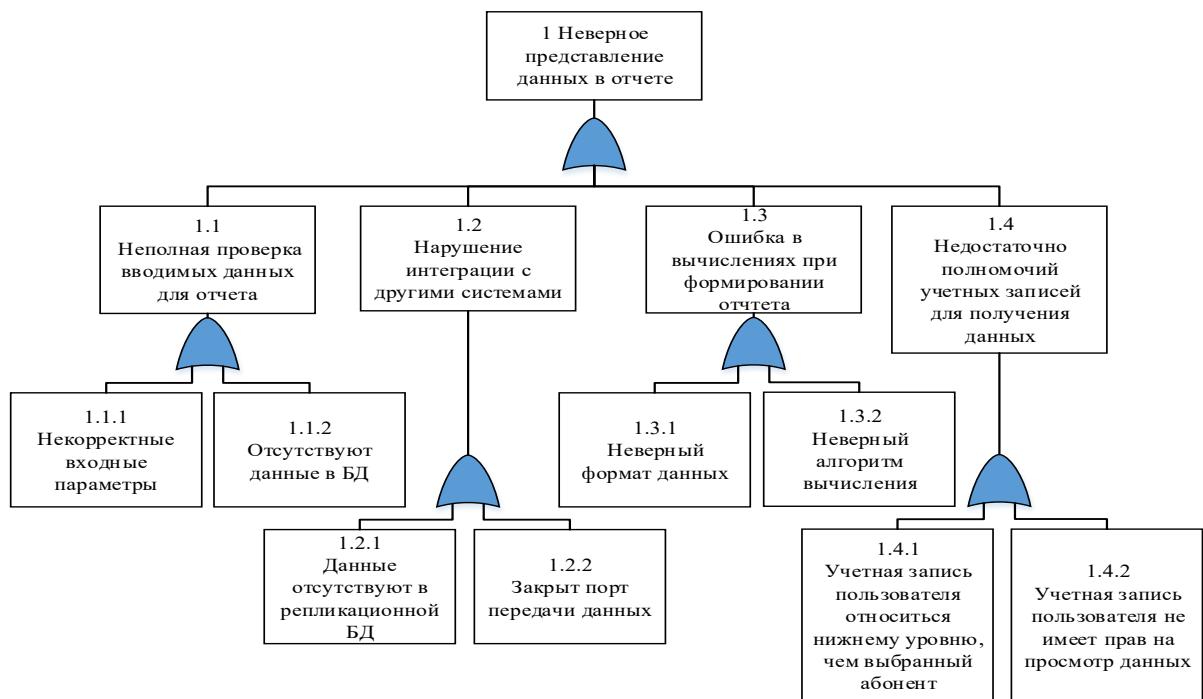


Рисунок 6.7 – FTA. Неверное представление данных в отчете

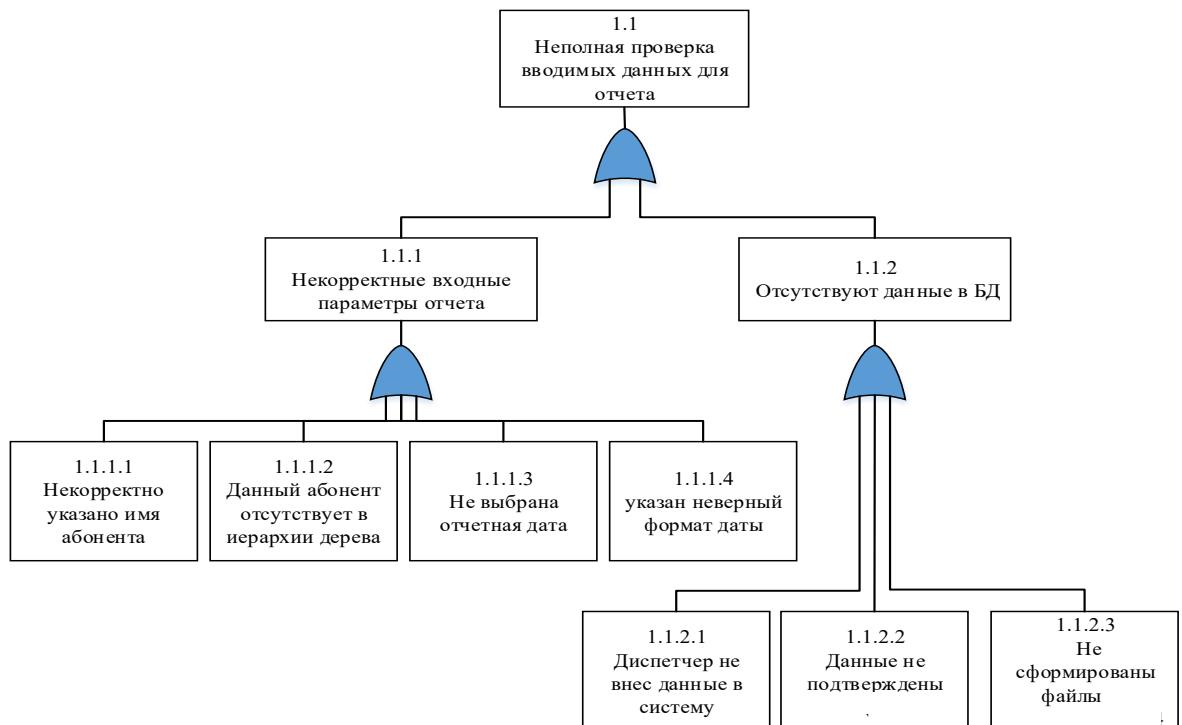


Рисунок 6.8 – FTA. Неполная проверка вводимых данных для отчета

В качестве примера реализации проактивного подхода на рисунке 6.9 представлена структура приложения по сопровождению системы, включающее в себя типовые инструкции (на основе полученных результатов *RCA* и *FTA*) по сопровождению локальной ИС.

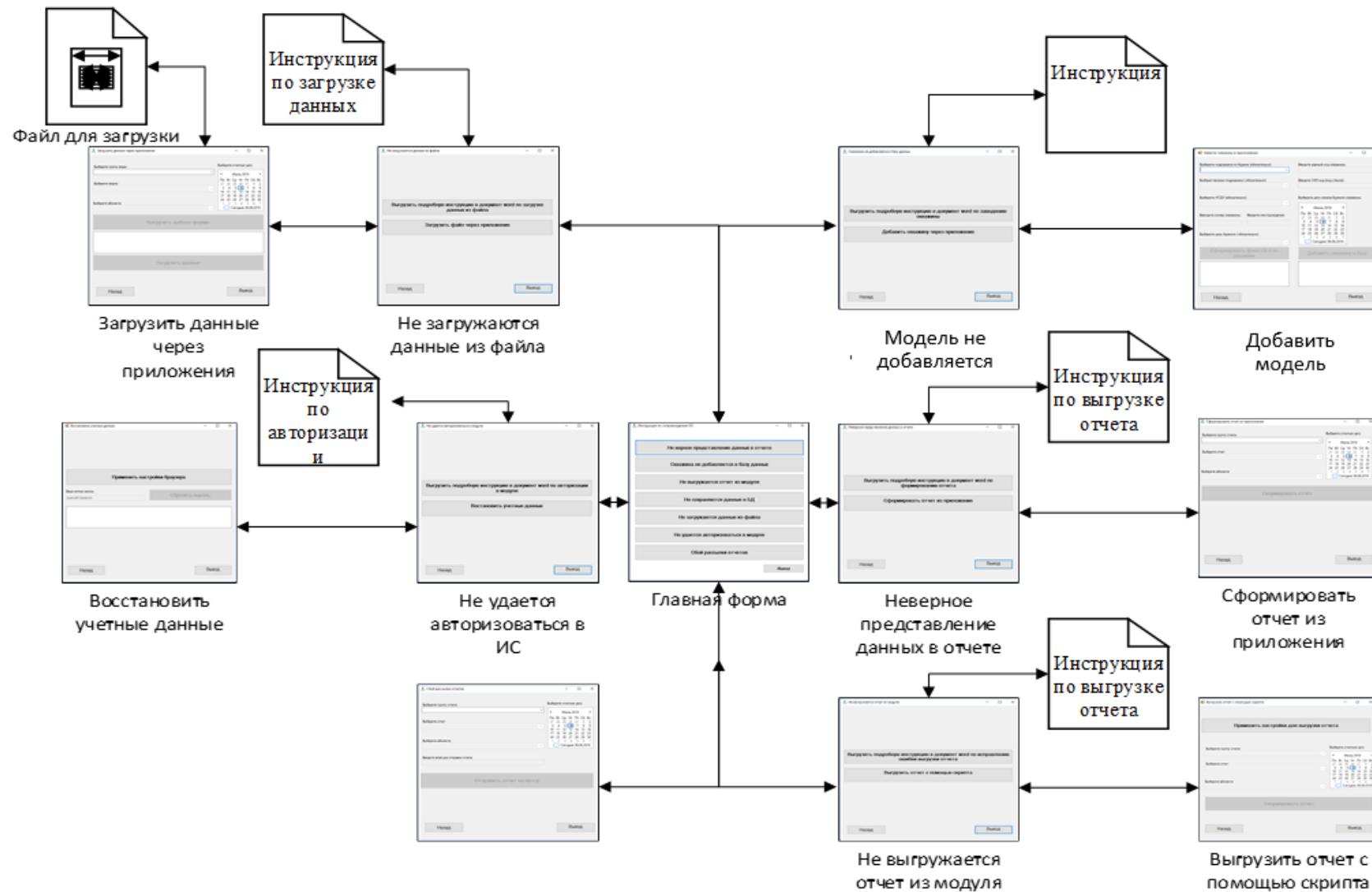


Рисунок 6.9 – Структура приложения по сопровождению

Главная форма приложения «Инструкция по сопровождению» (представлена на рисунке 6.10), включает типовые проблемы и быстрые переходы к инструкциям их решения.

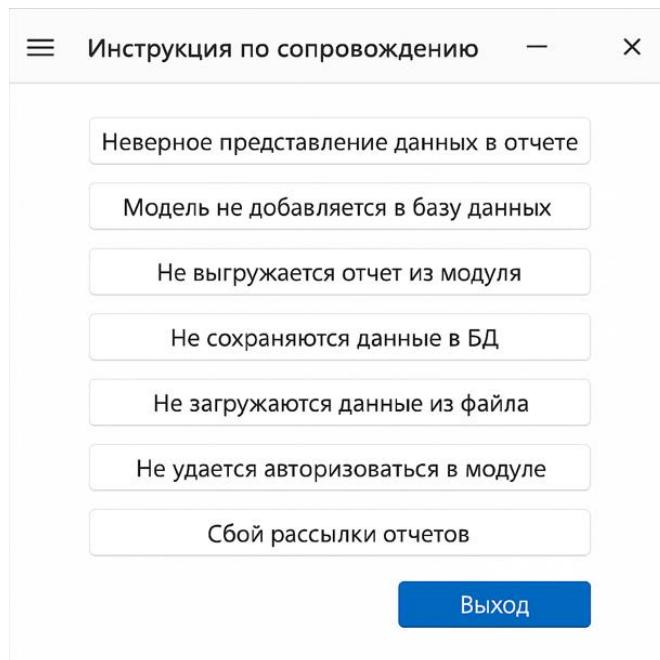


Рисунок 6.10 – Экранная форма приложения «Инструкция по сопровождению»

Главная форма является панелью навигации по типовым проблемам. Каждая кнопка открывает соответствующую инструкцию. Подробно элементы формы рассмотрены и пользовательские сценарии представлены в таблице 6.2 и 6.3.

Таблица 6.2 - Элементы формы

Элемент интерфейса	Назначение	Дальнейшие действия
Неверное представление данных в отчете	Ошибка структуры/визуализации отчета	Переход к инструкции по корректировке отчета
Модель не добавляется в базу данных	Сбой при сохранении модели в БД	Переход к инструкции по добавлению модели
Не выгружается отчет из модуля	Отчет не формируется или не сохраняется	Переход к инструкции по выгрузке отчета
Не сохраняются данные в БД	Ошибка записи данных	Переход к инструкции по проверке БД и логов
Не загружаются данные из файла	Сбой при импорте файлов	Переход к инструкции по загрузке данных
Не удается авторизоваться в модуле	Ошибка входа	Переход к инструкции по авторизации
Сбой рассылки отчетов	Отчеты не отправляются по рассылке	Переход к инструкции по устранению сбоев рассылки
Выход	Завершение работы приложения	Закрытие окна

Таблица 6.3 - Пользовательские сценарии: проблема -&gt; кнопка -&gt; инструкция

Проблема	Кнопка в главной форме	Действие пользователя	Инструкция
Неверное представление данных в отчете	Не верное представление данных в отчете	Проверить настройки формирования, пересоздать отчет	Инструкция по выгрузке отчета
Модель не добавляется в базу данных	Модель не добавляется в базу данных	Проверить параметры модели, повторить	Инструкция по добавлению модели
Отчет не выгружается из модуля	Не выгружается отчет из модуля	Повторить выгрузку, при необходимости использовать скрипт	Инструкция по выгрузке отчета
Данные не сохраняются в БД	Не сохраняются данные в БД	Проверить соединение с БД, логи	Инструкция по работе с БД
Данные не загружаются из файла	Не загружаются данные из файла	Проверить формат файла, шаблон	Инструкция по загрузке данных
Ошибка авторизации	Не удается авторизоваться в модуле	Проверить логин/пароль, восстановить учетные данные	Инструкция по авторизации
Сбой рассылки отчетов	Сбой рассылки отчетов	Проверить настройки SMTP/токенов, повторить рассылку	Инструкция по рассылке отчетов

Таким образом, у пользователя есть единый навигатор по ошибкам и решениям. Все инструкции доступны в текстовом и графическом виде. Рекомендуется использовать пошаговые указания внутри каждой инструкции.

Для того, чтобы оценить эффективность проведенных мероприятий необходимо рассчитать коэффициент готовности.

В теории надежности коэффициент готовности — это один из базовых количественных показателей [51], характеризующих способность системы находиться в работоспособном состоянии в произвольный момент времени, определяется по формуле:

$$K_{\Gamma} = \frac{T_{cp}}{T_{cp} + T_{в}} \quad (6.1)$$

где  $T_{cp}$  — среднее время работы до отказа (*MTTF* — *Mean Time To Failure*),

$T_{в}$  — среднее время восстановления (*MTTR* — *Mean Time To Repair*).

Рассчитаем коэффициент готовности ИС до и после проведения мероприятий по снижению ошибок.

1) Рассчитаем коэффициент готовности ИС 24×7 за месяц после проведения мероприятий по снижению ошибок.

*Известно:*

Месяц = 30 суток  $T_{\text{календ}} = 30 \times 24 = 720$  часов;

Инцидентов за месяц:  $N_0 = 40$ ;

Среднее время восстановления  $T_{\text{в0}} = 2,5$  часа

*Требуется:*

Оценить коэффициент готовности  $K_{\Gamma}$  информационной системы 24×7 за месяц до момента проведения мероприятий по снижению ошибок.

Итоговый простой за месяц (оценка):

$$D_0 = N_0 \times T_{\text{в}} = 40 \times 2,5 = 100 \text{ ч.}$$

$$T_{\text{ср0}} = \frac{720}{40} = 18 \text{ ч.}$$

В соответствии с (6.1):

$$K_{\Gamma 0} = \frac{18}{18+2,5} \approx 0,878 \quad (87,7\%);$$

2) Рассчитаем коэффициент готовности ИС 24×7 за месяц после проведения мероприятий по снижению ошибок (*FTA*, *RCA*, устранение коренных причин ошибок, внедрение типовых инструкций и др.).

*Известно:*

Месяц = 30 суток  $T_{\text{календ}} = 30 \times 24 = 720$  часов;

Инцидентов за месяц:  $N_1 = 12$ ;

Среднее время восстановления  $T_{\text{в1}} = 1$  час

*Требуется:*

Оценить коэффициент готовности  $K_{\Gamma}$  информационной системы 24×7 за месяц после проведения мероприятий по снижению ошибок.

Итоговый простой за месяц (оценка):

$$D_1 = N_1 \times T_{\text{в}} = 12 \times 1 = 12 \text{ ч.}$$

$$T_{\text{ср1}} = \frac{720}{12} = 60 \text{ ч.}$$

В соответствии с (6.1):

$$K_{\Gamma 1} = \frac{60}{60+1} \approx 0,984 \quad (98,4\%)$$

3) Полученные результаты оценки надежности ИС запишем в таблицу 6.4.

Таблица 6.4 - Результаты оценки надёжности ИС

Показатель	До внедрения мероприятий	После внедрения мероприятий	Изменение
Инциденты за месяц, $N$	40	12	- 70 %
Среднее время работы до отказа, $T_{cr}$	18	60	×3,33
Среднее время восстановления, $T_b$	2,5	1	-60 %
Итоговый простой за месяц	100	12	- 88 ч
Коэффициент готовности, $K_G$	≈ 0,878	≈ 0,984	10,6 п.п

Использование коэффициента готовности в качестве интегрального показателя функциональной надежности позволяет объединить результаты *RCA* и *FTA* в единую метрику, отражающую фактическое состояние ИС. Такой подход дает возможность сопоставить состояние системы до и после внедрения мероприятий.

Результаты расчётов коэффициента готовности ИС показали, что после внедрения организационно-технических мероприятий количество отказов сократилось более чем в три раза, среднее время восстановления уменьшилось на 60 %, а коэффициент готовности возрос с 87,8 % до 98,4 %. Данный результат подтверждает эффективность комплексного подхода, включающего как устранение коренных причин отказов, так и внедрение типовых инструкций сопровождения.

Таким образом, выявлены факторы функциональной надежности ИС при формировании единого информационного пространства компании. Проанализированы проблемные ситуации, возникающие при работе с локальными ИС на основе поступивших запросов в службу поддержки. Выявлены коренные причины, способствующие возникновению нежелательного события в системе. Полученные результаты легли в основу типовых инструкций по сопровождению локальных ИС. Полученные результаты демонстрируют возможность интеграции реактивного и проактивного подходов. Результаты расчётов коэффициента готовности ИС подтвердили высокую эффективность принятых мер.

## 6.2 Методика формальной оценки возможности реализации проекта

Настоящий раздел диссертационной работы посвящен разработке методики формальной оценки возможности реализации проекта. Основу методики составляют теоретические результаты, полученные в разделах 5.2 и 5.3.

Полученные научные результаты апробированы в одной из компаний группы «ЛУКОЙЛ». ЛУКОЙЛ - одна из крупнейших вертикально интегрированных нефтегазовых компаний в мире, на долю которой приходится более 2% мировой добычи нефти и около 1% доказанных запасов углеводородов<sup>17</sup>. Бизнес-модель компании представлена на рисунке 6.11.



Рисунок 6.11 – Бизнес-модель компании «ЛУКОЙЛ»<sup>18</sup>

Группа «ЛУКОЙЛ» реализует политику в области устойчивого развития. Фрагмент из отчета за 2023 год представлен на рисунке 6.12.

<sup>17</sup> Источник - <https://lukoil.ru/> -официальный сайт компании «Лукойл»

<sup>18</sup> Источник - <https://lukoil.ru/Company/BusinessOperation> -официальный сайт компании «Лукойл»

## Политика Группы «ЛУКОЙЛ» в области устойчивого развития



## Наш вклад в Цели устойчивого развития ООН в 2023 году



Рисунок 6.12 – Фрагмент отчета об устойчивом развитии группы «ЛУКОЙЛ»<sup>19</sup>

<sup>19</sup> Источник – Отчет об устойчивом развитии группы «ЛУКОЙЛ» за 2023г. <https://lukoil.ru/FileSystem/9/666712.pdf> [93]

Применение цифровых технологий и автоматизация бизнес-процессов оказывают существенное воздействие на непрерывность, надежность и безопасность производств, позволяют снижать операционные затраты и увеличивать эффективность деятельности в основных бизнес-сегментах, улучшая экономические показатели [93]. Ключевые приоритеты Компании - обеспечение надежного функционирования информационно-технологического (ИТ) ландшафта<sup>20</sup>.

В отчете представлены данные о реализуемых программах. Одной из программ является – «Функциональная программа развития по информационно-технологическому обеспечению Группы «ЛУКОЙЛ».

Данная программа рассчитана на 2023-2025гг, сфокусирована на поддержании непрерывности бизнес-процессов и на обеспечении надежности и независимости ИТ-инфраструктуры. В связи с изменившимися внешними условиями осуществляется переход на новые платформы с учетом доступности применявшихся ранее решений.

Компания продолжает совершенствовать систему управления информационно-технологическим обеспечением. Основными задачами являются обеспечение технологической независимости и динамического масштабирования ключевых элементов ИТ-ландшафта. ИС являются частью ИТ-ландшафта. К настоящему времени в рамках программы реализован ряд проектов по переходу на независимые решения в части корпоративной ИТ-инфраструктуры.

Полученные в диссертационной работе научные результаты (раздел 5.1 - 5.3) положены в основу методики формальной оценки возможности реализации проекта.

В рамках «Функциональной программы развития по информационно-технологическому обеспечению» использование предлагаемой методики будет способствовать повышению обоснованности параметров проекта.

---

<sup>20</sup> ИТ-ландшафт – все приложения, используемые в системной среде, а также все компоненты инфраструктуры сети, данные, которыми необходимо управлять, и интерфейсы между компонентами [93].

Предлагаемая методика позволяет на предпроектной стадии оценить предполагаемую удовлетворенность заказчика результатами проекта, а исполнителей – ходом проекта в зависимости от бюджета и ограничений на длительность проекта с учетом опыта, полученного от ранее реализованных проектов. Схематично последовательность этапов методики, представлена на рисунке 6.13.

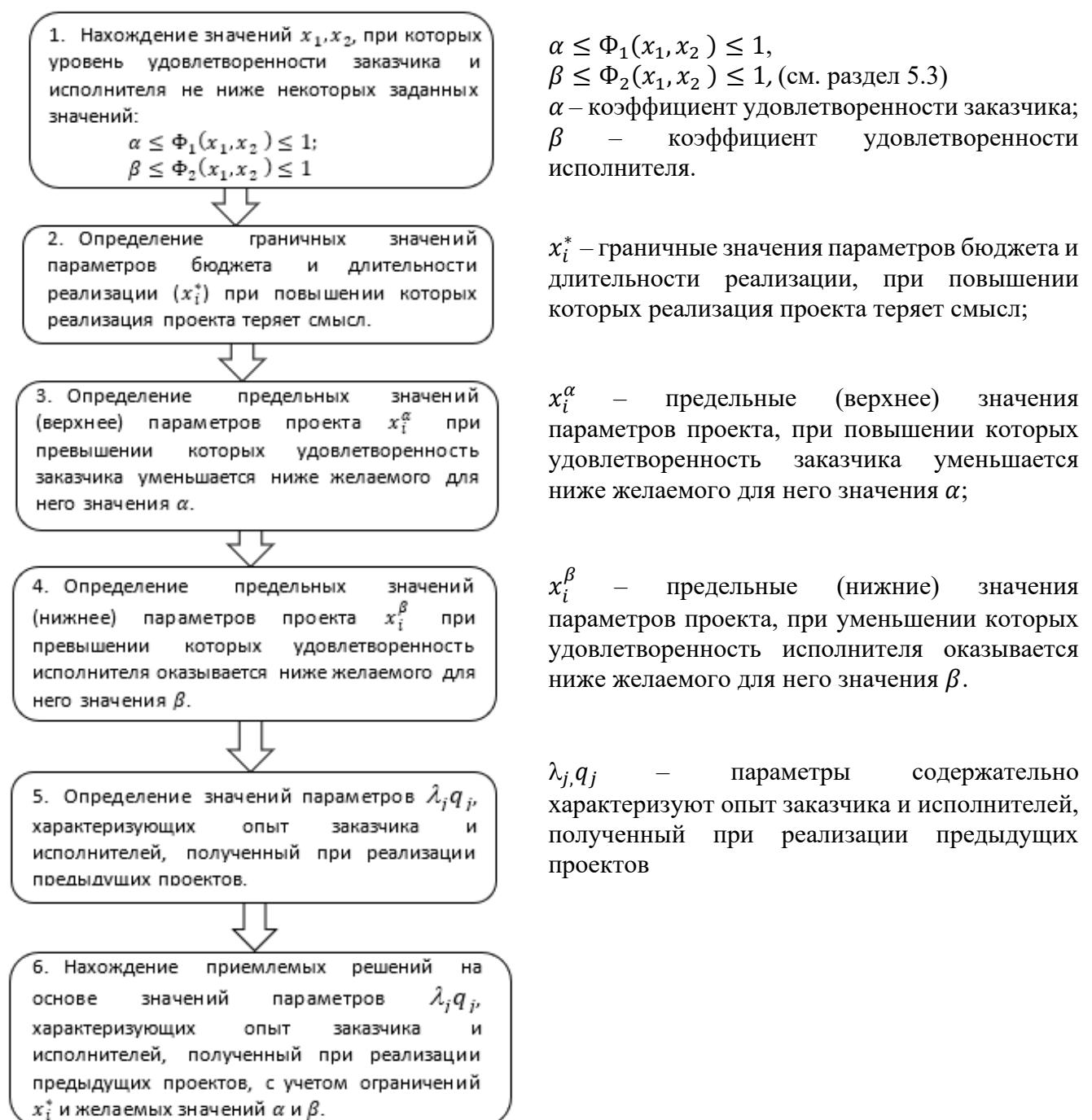


Рисунок 6.13 – Основные этапы методики формальной оценки  
возможности реализации проекта

Методика включает в себя следующие этапы:

1. Нахождение значений  $x_1, x_2$  (бюджет и длительность реализации проекта) при которых уровень удовлетворенности заказчика и исполнителя не ниже некоторых заданных значений.
2. Определение граничных значений параметров бюджета и длительности реализации при повышении которых реализация проекта теряет смысл.
3. Определение предельных значений (верхнее) параметров проекта при превышении которых удовлетворенность заказчика уменьшается ниже желаемого для него значения.
4. Определение предельных значений (нижнее) параметров проекта при превышении которых удовлетворенность исполнителя оказывается ниже желаемого для него значения.
5. Определение значений параметров, характеризующих опыт заказчика и исполнителей, полученный при реализации предыдущих проектов.
6. Нахождение приемлемых решений на основе значений параметров, характеризующих опыт заказчика и исполнителей, полученный при реализации предыдущих проектов, с учетом ограничений и желаемых значений параметров проекта.

В качестве демонстрационного примера рассмотрим формальную оценку возможности реализации проекта ИС (на примере ИС для организации комплексной системы безопасности).

В качестве заказчика рассмотрим одну из компаний группы «ЛУКОЙЛ», «ЛУКОЙЛ-Центрнефтепродукт», направления деятельности связано со сбытом нефтепродуктов (выделено пунктирной линией на рисунке 6.11) и относящиеся к ним подразделения по информационной и корпоративной безопасности (выделено пунктирной линией на рисунке 6.14).

«ЛУКОЙЛ-Центрнефтепродукт» - один из ведущих операторов оптово-розничной торговли нефтепродуктами в Москве, Нижнем Новгороде, Московской, Владимирской, Калужской, Липецкой, Брянской, Нижегородской, Орловской, Рязанской, Смоленской, Тамбовской, Тверской, Тульской, Ярославской областях,

Республике Марий Эл, Республике Мордовия, Республике Татарстан и Чувашской Республике. Структура предприятия включает 7 нефтебаз и более 640 автозаправочных станций.

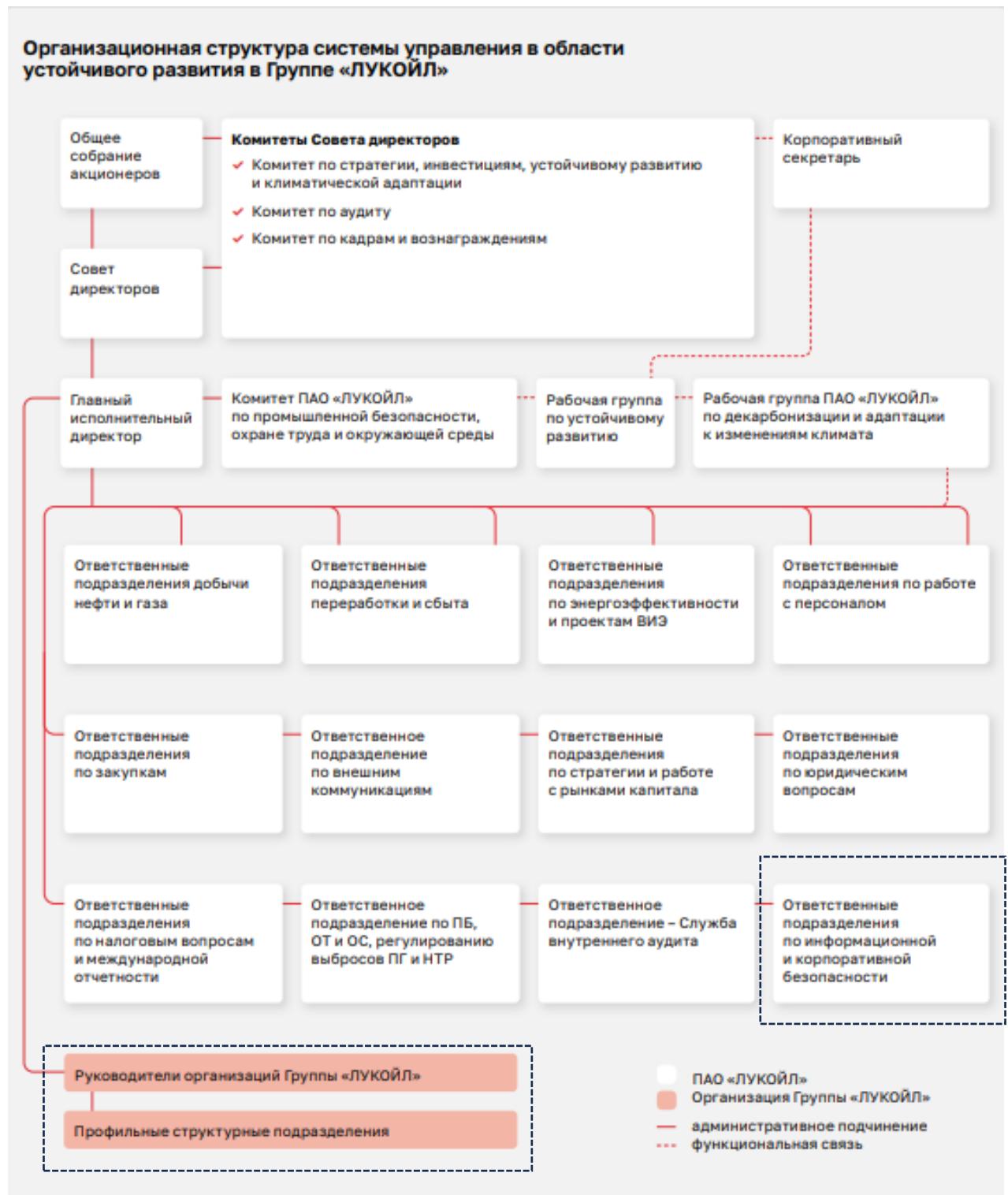


Рисунок 6.14 – Фрагмент организационной структуры<sup>21</sup>

<sup>21</sup> Источник – Отчет об устойчивом развитии группы «ЛУКОЙЛ» за 2023г. <https://lukoil.ru/FileSystem/9/666712.pdf> [93]

В качестве исполнителя рассмотрим разработчиков программной платформы «Интеллект». «Интеллект» – многофункциональная открытая программная платформа, предназначенная для создания комплексных систем безопасности любого масштаба.

Система безопасности на базе программной платформы «Интеллект» объединяет видеонаблюдение, охранно-пожарную сигнализацию, систему охраны периметра, систему контроля и управления доступом, аудиоконтроль в согласованно работающую инфраструктуру.

Таким образом, на основе программной платформы «Интеллект» комплекс различных систем безопасности превращается в единую информационную среду. На рисунке 6.15 представлен пример распределенной архитектуры ИС на основе платформы «Интеллект».

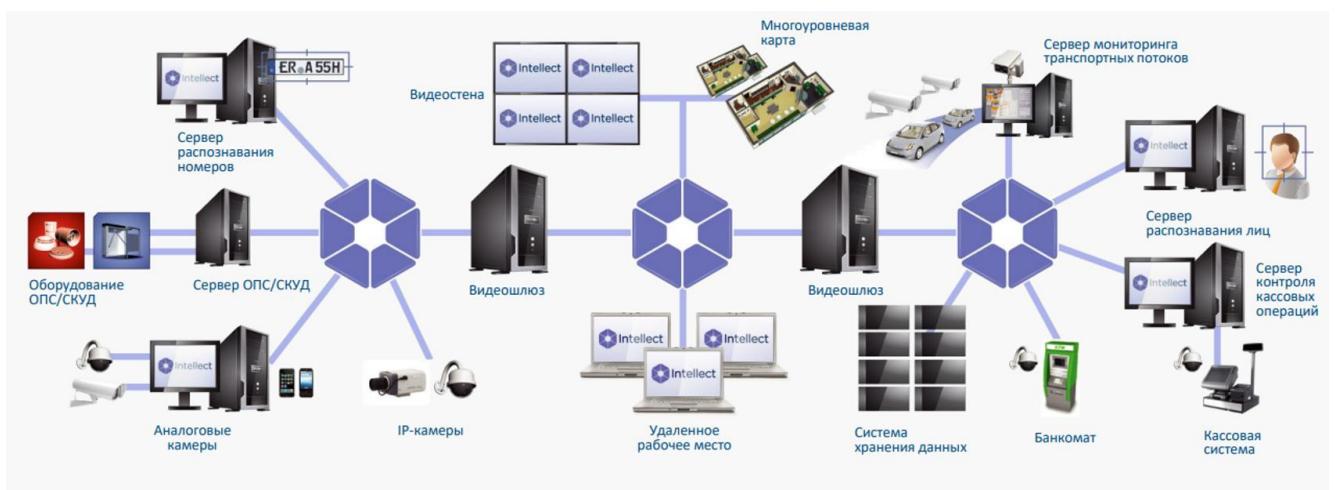


Рисунок 6.15 – Пример распределенной архитектуры ИС на основе платформы «Интеллект»<sup>22</sup>

Распределенная архитектура платформы «Интеллект» позволяет решать весь спектр задач обеспечения комплексной безопасности на этапах сбыта нефтепродуктов.

<sup>22</sup> Источник – <https://www.itv.ru/> официальный сайт компании ITV (российский разработчик программного обеспечения для систем безопасности и видеонаблюдения)

С целью повышения обоснованности параметров проекта проведем формальную оценку возможности реализации проекта на основе полученных теоретических результатов.

В качестве исходных данных воспользуемся накопленными данными (фрагмент логической модели базы данных представлен на рисунке 6.16).

Модель поддерживает учет проектов, их календарных параметров, бюджета, состава команды, взаимоотношений с заказчиками и исполнителями, а также фиксацию статусов/отчетов и событий по проекту.

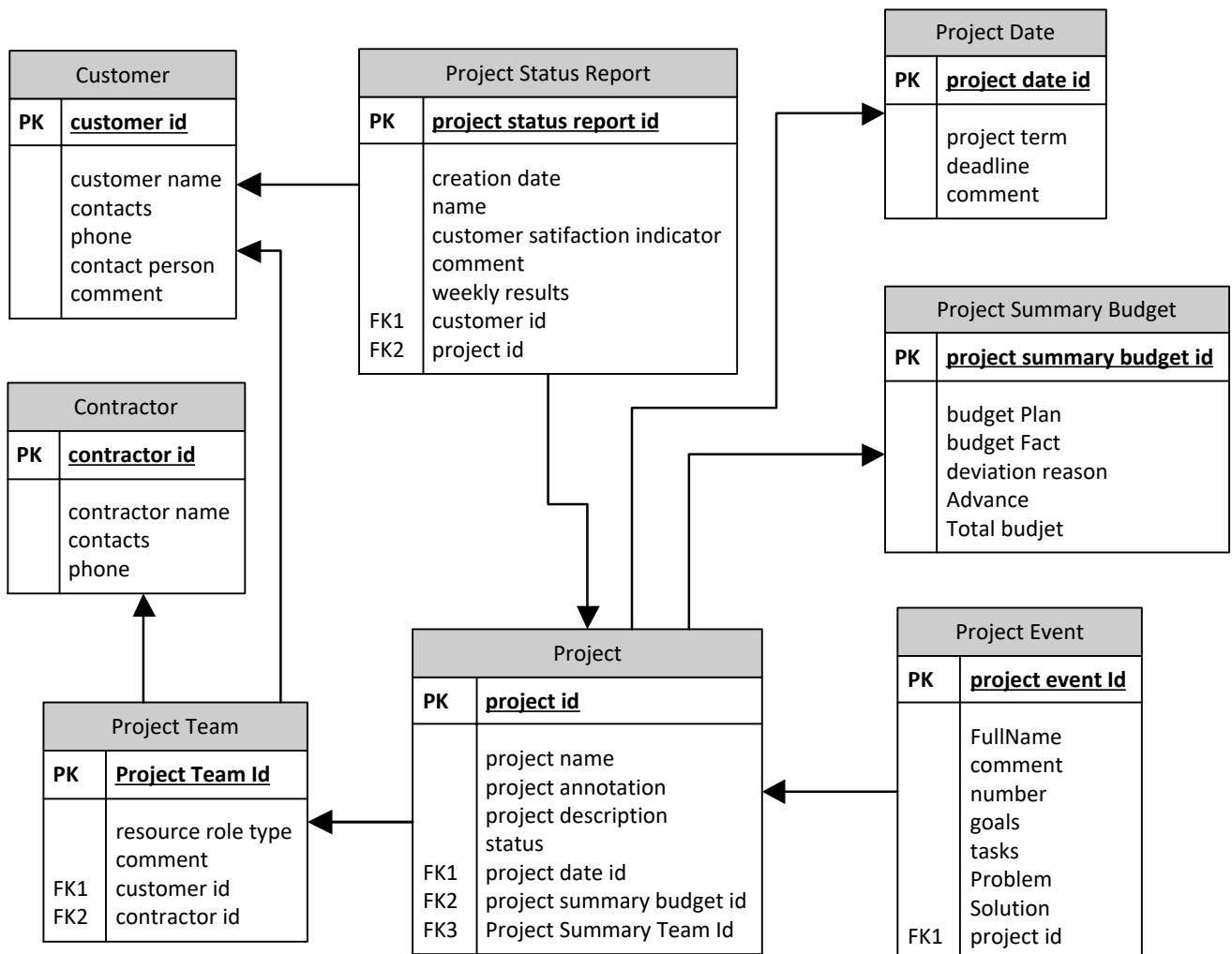


Рисунок 6.16 – Фрагмент логической модели БД

Бизнес-правила:

- 1) обязательность связей проекта: при создании Project должны быть заданы валидные ссылки на календарь (project\_date\_id) и бюджет (project\_summary\_budget\_id);

2) уникальность названия: `project_name` рекомендуется ограничить уникальным индексом в пределах активных проектов;

3) историчность: при изменении сроков и/или бюджета новая версия записывается в соответствующие таблицы, а в `Project` обновляется ссылка на актуальную запись;

4) согласованность бюджета: `total_budget` должен быть согласован с `budget_plan/advance`; при наличии `budget_fact` допустимы неотрицательные отклонения, причины фиксируются в `deviation_reason`;

5) статус-отчеты: для каждого проекта допускается не более одного статус-отчета за единицу отчетного периода (например, неделю), периодность задается регламентом;

6) события: поля `problem/solution` используются для причинно-следственной фиксации; рекомендуются справочники типов событий и статусов обработки.

В таблице 6.5 представлено описание сущностей и атрибутов, а таблице 6.6 описание связей между сущностями.

Таблица 6.5 – Сущности и атрибуты

Сущность	PK	FK	Атрибуты	Назначение
Customer	<code>customer_id</code>	–	<code>customer_name,</code> <code>contacts,</code> <code>phone,</code> <code>contact_person,</code> <code>comment</code>	Заказчик проекта
Contractor	<code>contractor_id</code>	–	<code>contractor_name,</code> <code>contacts,</code> <code>phone</code>	Подрядчик проекта
Project Team	<code>project_team_id</code>	<code>customer_id,</code> <code>contractor_id</code>	<code>resource_role_type,</code> <code>comment</code>	Участники команды проекта
Project Date	<code>project_date_id</code>	–	<code>project_term,</code> <code>deadline,</code> <code>comment</code>	Сроки проекта
Project Summary Budget	<code>project_summary_budget_id</code>	–	<code>budget_plan,</code> <code>budget_fact,</code> <code>deviation_reason,</code> <code>advance,</code> <code>total_budget</code>	Сводный бюджет
Project	<code>project_id</code>	<code>project_date_id,</code> <code>project_summary_budget_id,</code> <code>project_team_id</code>	<code>project_name,</code> <code>project_annotation,</code> <code>project_description,</code> <code>status</code>	Центральная сущность

Сущность	PK	FK	Атрибуты	Назначение
Project Status Report	project_status_report_id	customer_id, project_id	creation_date, name, customer_satisfaction_indicator, comment, weekly_results	Отчет по статусу проекта
Project Event	project_event_id	project_id	full_name, comment, number, goals, tasks, problem, solution	События и инциденты проекта

Таблица 6.6 - Связи

От	К	Тип связи	Кардинальность	Описание
Project → Project Date	project_date_id	FK	1:1	Проект связан с одной записью сроков
Project → Project Summary Budget	project_summary_budget_id	FK	1:1	Один бюджет на проект
Project → Project Team	project_team_id	FK	1:1	Ответственная команда
Status Report → Project	project_id	FK	M:1	Множество отчетов для одного проекта
Status Report → Customer	customer_id	FK	M:1	Отчет связан с заказчиком
Project Event → Project	project_id	FK	M:1	Множество событий для проекта
Project Team → Customer	customer_id	FK	M:1	Член команды от заказчика
Project Team → Contractor	contractor_id	FK	M:1	Член команды от подрядчика

Таким образом, представленная модель данных выступает как базовый источник данных для применения методики оценки возможности реализации проекта.

В таблице 6.7 представлены исторические сведения о двадцати проектах, схожих с тем, параметры которого оцениваются.

Таблица 6.7 – Исторические сведения о проектах

n/n	$x_1$	$x_2$	$z_1(x_1, x_2)$	$z_2(x_1, x_2)$
1	2,5	1,5	0,9	0,95
2	1	3	0,9	0,9
3	2	1	1	0,8
4	3,5	2	0,7	1
5	2	5	0,6	1
6	3	5	0,4	1
7	2,5	2,5	0,7	1
8	2	3	0,75	1
9	0,5	0,5	1	0,3
10	5	5	0,3	1
11	5	5	0,3	1
12	2,5	2,5	0,7	1
13	3	3	0,75	1
14	0,5	0,5	1	0,3
15	2,5	1,5	0,9	0,95
16	1	3	0,9	0,9
17	3	5	0,4	1
18	5	5	0,3	1
19	2	1	1	0,8
20	3,5	2	0,7	1

*Этап 1.* Реализация первого этапа методики «Формальной оценки возможности реализации проекта» включает в себя:

Нахождение значений  $x_1, x_2$ , при которых уровень удовлетворенности заказчика и исполнителя не ниже некоторых заданных значений:

$$\alpha \leq \Phi_1(x_1, x_2) \leq 1;$$

$$\beta \leq \Phi_2(x_1, x_2) \leq 1;$$

определяется в соответствии с разделом 5.3

Желаемые значения удовлетворенности составляют:  $\alpha=0,9$ ;  $\beta=0,95$ .

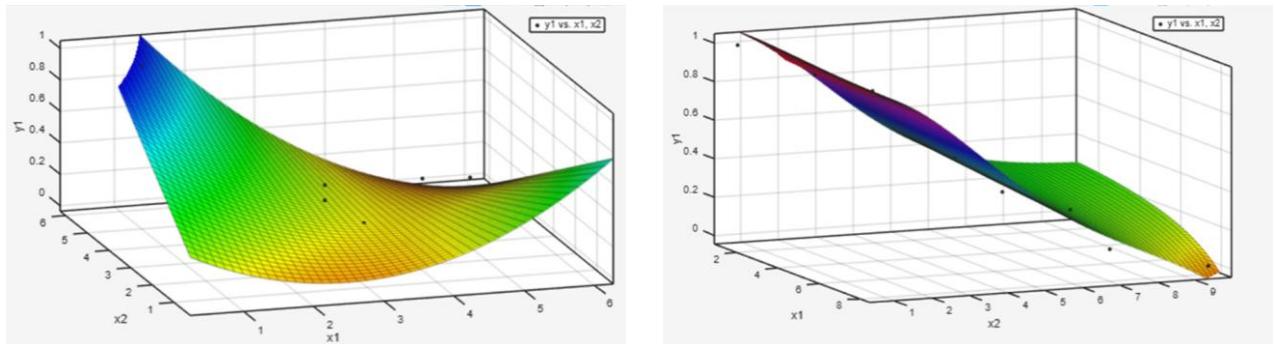
*Этап 2.* Определение граничных значений параметров бюджета и длительности реализации ( $x_i^*$ ) при повышении которых реализация проекта теряет смысл.

Граничные значения составляют:  $x_i^* = 5$ , ( $i=1,2$ ).

*Этап 3.* Определение предельных значений (верхнее) параметров проекта  $x_i^\alpha$  при превышении которых удовлетворенность заказчика уменьшается ниже желаемого для него значения  $\alpha$ .

Для фиксированного бюджета определяется максимальная длительность, при которой сохраняется удовлетворенность заказчика, аналогично - для фиксированной длительности определяется максимальный бюджет.

На рисунке 6.16 представлена область допустимых значений для заказчика.



$$\Phi_1(x_1, x_2)$$

Рисунок 6.17 – Область допустимых значений для заказчика

*Этап 4.* Определение предельных значений (нижнее) параметров проекта  $x_i^\beta$  при превышении которых удовлетворенность исполнителя оказывается ниже желаемого для него значения  $\beta$ .

Для фиксированного бюджета определяется минимальная длительность, обеспечивающая требуемый уровень удовлетворенности исполнителя, аналогично - для фиксированной длительности находится минимально необходимый бюджет.

На рисунке 6.17 представлена область допустимых значений для исполнителя.

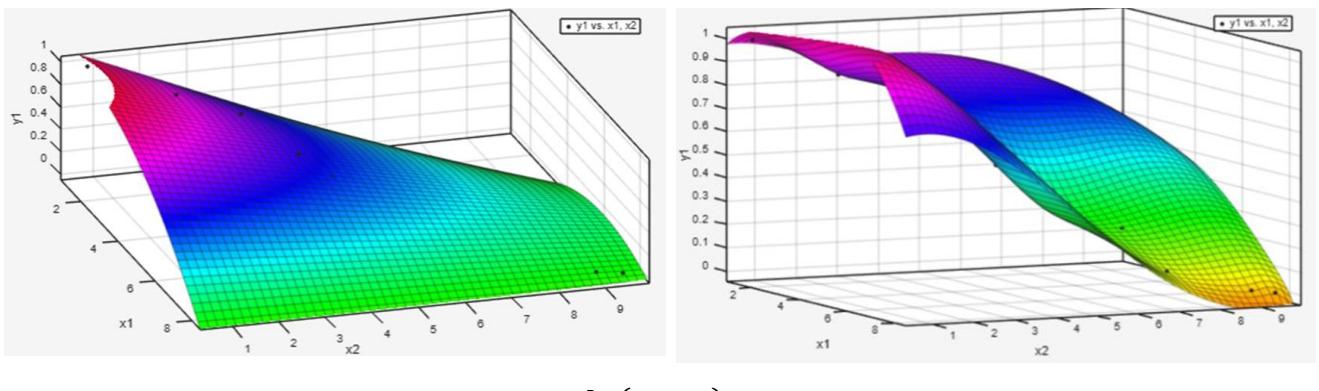


Рисунок 6.18 – Область допустимых значений для исполнителя

Реализация проекта возможна лишь в условиях сбалансированности запросов к параметрам проекта заказчика и исполнителя (выражается значениями  $\alpha$  и  $\beta$ ) (графическая иллюстрация ситуации представлена на рисунке 5.7 в разделе 5.3).

*Этап 5.* Определение значений параметров  $\lambda_j q_j$ , характеризующих опыт заказчика и исполнителей, полученный при реализации предыдущих проектов.

Сопоставимость персонального опыта заказчика и исполнители, накопленного ими при реализации прежних проектов выражается посредством функциональных зависимостей  $\varphi_{ij}(x_i)$ ,  $i, j = 1, 2$  (подробное описание функциональных зависимостей представлено в разделе 5.3, зависимости (5.16) и (5.17)). Исходные данные для расчетов представлены в таблице 6.2.

В результате решения задач (5.18) и (5.19) методом деформируемого многогранника, получаем значения коэффициентов, характеризующих опыт заказчика и исполнителей:

$$\lambda_1 = \lambda_2 = 0.4; q_1 = q_2 = 2.$$

*Этап 6.* Нахождение приемлемых решений на основе значений параметров  $\lambda_j q_j$ , характеризующих опыт заказчика и исполнителей, полученный при реализации предыдущих проектов (определяется на этапе 5), с учетом ограничений  $x_i^*$  (определяется на этапе 2) и желаемых значений  $\alpha$  и  $\beta$  (определяется на этапе 1).

Графические модели, характеризующие возможность нахождения приемлемых решений при различных значениях удовлетворенности заказчика и исполнителя, представлены на рисунках 6.18 и 6.19.

Рисунки предназначены для демонстрации ситуаций, когда нахождения приемлемого для заказчика и исполнителя решения о характеристиках проекта исходя из определяемыми как заказчиком, так и исполнителем ограничений на уровень их удовлетворенности исходом проекта возможно (рисунок 6.18) и невозможно (рисунок 6.19).

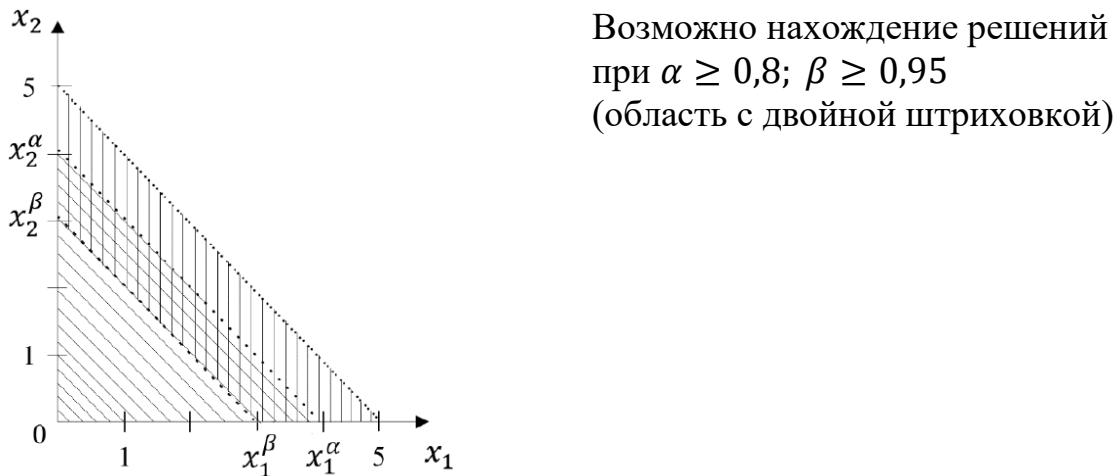


Рисунок 6.19 – Возможность нахождения приемлемых решений

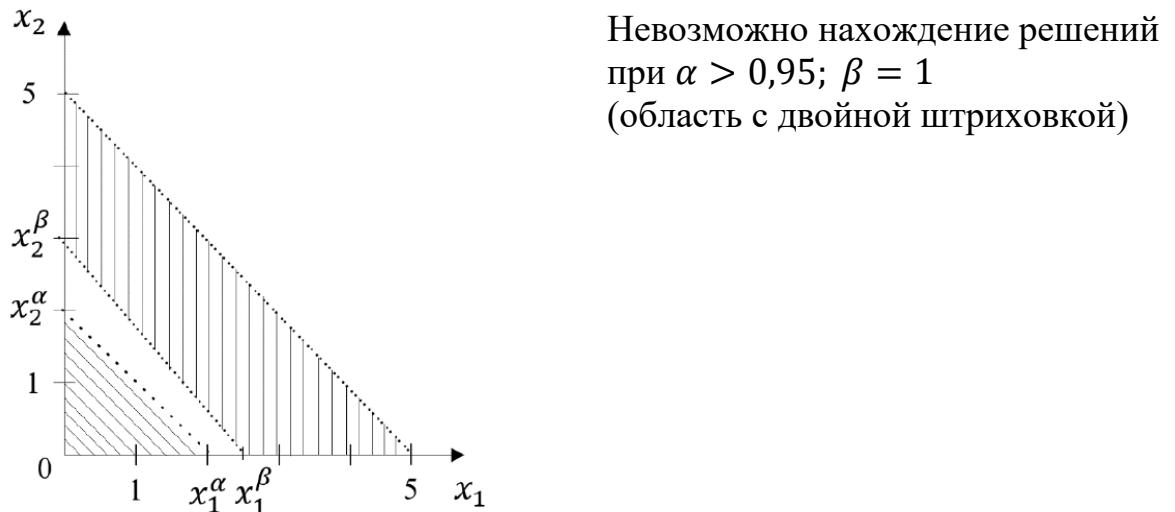


Рисунок 6.20 – Возможность нахождения приемлемых решений

отсутствует

Формализация процедуры нахождения области возможных решений ускоряет процесс формирования консолидированного решения при определении базовых характеристик проекта, а также облегчает взаимодействие между заказчиком и исполнителем.

Кроме того, возможно развитие модульной версии методики, где отдельные этапы (например, учет опыта или определение предельных значений) могут применяться автономно в качестве инструментов поддержки принятия решений. Объектно-ориентированные модели в нотации UML, описывающие логику работы, представлены в приложении В.

Однако необходимо отметить ограничения предложенной методики, которые необходимо учитывать при ее практическом применении:

- 1) корректность расчетов зависит от полноты и качества данных о ранее реализованных проектах;
- 2) используемые функции  $\Phi_1(x_1, x_2)$ ;  $\Phi_2(x_1, x_2)$  являются аппроксимациями субъективных оценок заказчиков и исполнителей, при изменении состава участников проекта параметры моделей требуют пересмотра.

Таким образом, методика формальной оценки возможности реализации проекта представляет собой один из инструментов предпроектного анализа. Методику рекомендуется применять руководителю или менеджеру проекта ИС на предпроектной стадии.

Оценку эффективности разработанной методики проведем по параметру оценка затраченного времени на предпроектную стадию. Результаты такой оценки дают возможность определить, какой процент времени затрачивался на формирование бюджета и сроков реализации проекта до и после внедрения методики формальной оценки его осуществимости.

С целью количественной оценки временных затрат были построены диаграммы Ганта<sup>23</sup>. Для определения длительности каждого этапа на представленных диаграммах был проведен опрос экспертов, которые ранее сталкивались с проблемными ситуациями, связанными с определением бюджета и сроков реализации проектов.

Согласно экспертным оценкам, продолжительность предпроектной стадии, предшествующей внедрению разработанной методики, находится в диапазоне

---

<sup>23</sup> Диаграмма Ганта (англ. *Gant chart*) - это тип столбчатой диаграммы, иллюстрирующий график реализации проекта. Диаграмма Ганта может включать даты начала и окончания задач, процент исполнения задач, зависимости между задачами, ресурсы проекта и прочее [115].

[26; 30] рабочих дней. На основе анализа эмпирических данных было определено, что среднее значение длительности указанного этапа составляет 27 рабочих дня, соответствует критическому пути, представленному на рисунке 6.20.

После внедрения разработанной методики предпроектная стадия составила 23 рабочих дня за счет формализации процедуры нахождения области возможных решений. Критический путь представлен на рисунке 6.21.

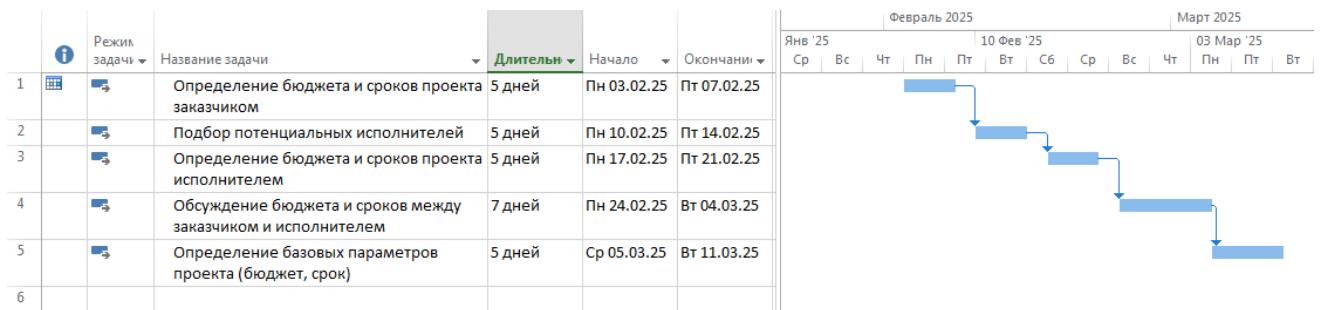


Рисунок 6.21 – Диаграмма Ганта, соответствующая предпроектной стадии до внедрения методики

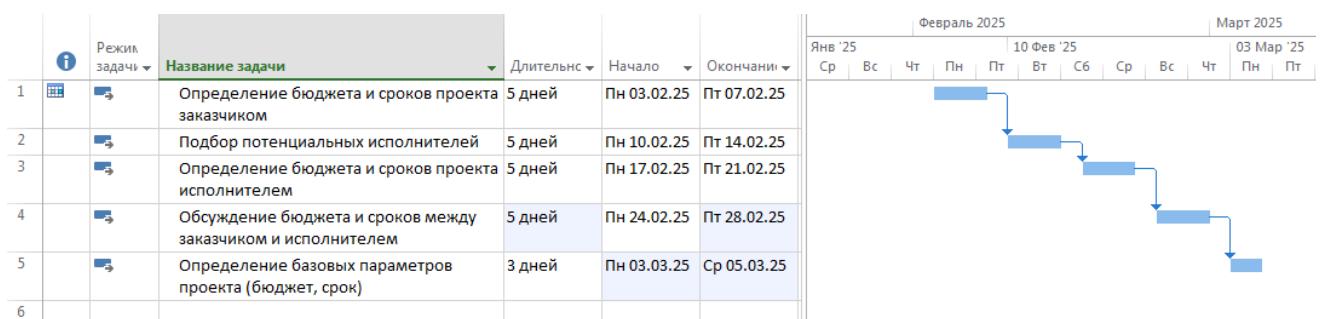


Рисунок 6.22 – Диаграмма Ганта, соответствующая предпроектной стадии после внедрения методики

Таким образом, методика формальной оценки возможности реализации проекта на предпроектной стадии позволяет оценить предполагаемую удовлетворенность заказчика результатами проекта, а исполнителей – ходом проекта в зависимости от бюджета и ограничений на длительность проекта с учетом опыта, полученного от ранее реализованных проектов.

Это позволяет получить экономический эффект за счет сокращения сроков предпроектной стадии (на 14%) и повышения обоснованности параметров проекта. Разработанная методика реализует барьерный подход к предотвращению организационных дефектов на предпроектной стадии.

## 6.3 Методика и программное обеспечение для анализа моделей проектируемых компонентов информационных систем по критериям функциональной надежности

В данном параграфе рассматриваются вопросы разработки методики и прототипа программного обеспечения, основу которого составляет предложенный автором метод структурного моделирования последствий отказов на основе аппарата схем сопряжения и таблиц истинности, предложенный в разделе 4.3.

Подробно описываются этапы проектирования и реализации прототипа, начиная с анализа требований и заканчивая созданием рабочей версии, которая демонстрирует основные функции программного обеспечения.

В соответствии с разработанным методом структурного моделирования последствий отказов программное обеспечение должно выполнять следующие функции:

- построение схемы сопряжения агрегатов;
- формирование таблиц истинности;
- построение на основе схемы сопряжения структурной модели отказов (*FTA*);
- выявление коренных причин отказов (*RCA*).

Данный функционал направлен на обеспечение анализа отказов, начиная с визуализации взаимосвязей между компонентами системы и заканчивая исследованием причин возникновения отказов.

С целью описания логики работы программы и наглядного представления структуры, поведения и взаимодействия компонентов системы, разработаны объектно-ориентированные модели. В качестве CASE-средства проектирования *UML*-диаграмм использовалось инструментальное средство *draw.io*, получившее широкое распространение благодаря своей простоте, удобству и наглядности.

Далее представлены диаграммы с пояснениями, которые помогут понять логику проектирования, ключевые особенности и взаимосвязи между компонентами.

Диаграмма вариантов использования (*Use Case*), представленная на рисунке 6.23 отображает взаимодействие между пользователями (акторами) и системой, связи между акторами и вариантами использования показывают, какие функции системы доступны каждому из участников. Диаграмма *Use Case* разработана для описания основных функций системы и взаимодействия пользователей с ней.

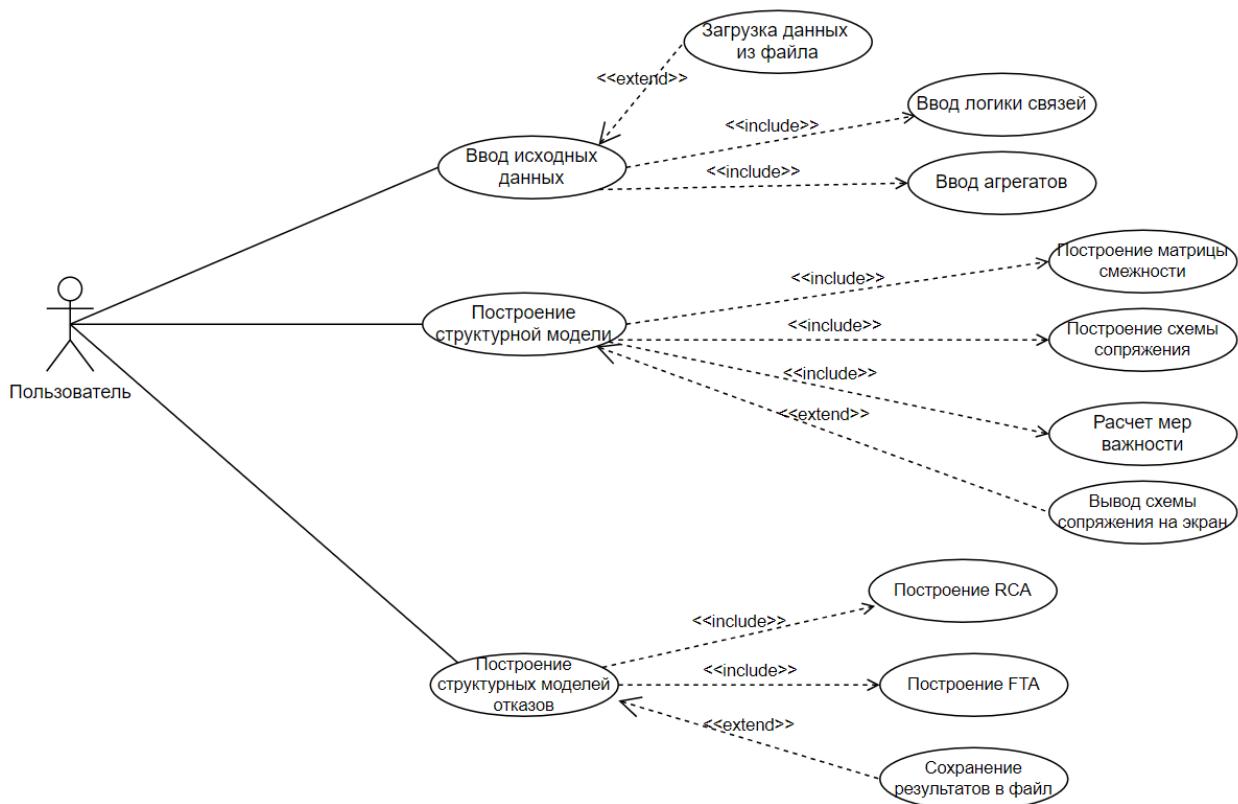


Рисунок 6.23 – Диаграмма вариантов использования (*Use Case*)

Пояснения к диаграмме вариантов использования (рисунок 6.23):

- 1) Диаграмма ориентирована на одного основного актора «Пользователь», который инициирует три ключевых ветви работы: ввод данных, построение структурной модели и построение моделей отказов.
- 2) Отношение *include* означает обязательное выполнение подфункции в составе базового варианта использования. Так, «Ввод логики связей» и «Ввод агрегатов» обязательно сопровождают «Ввод исходных данных», «Построение матрицы смежности» и «Построение схемы сопряжения» – неотъемлемые этапы «Построения структурной модели».

3) Отношение *extend* отражает опциональные действия, запускаемые при наличии соответствующих условий или решения пользователя: «Загрузка данных из файла» расширяет «Ввод исходных данных»; «Расчет мер важности» расширяет «Построение структурной модели»; «Сохранение результатов в файл» расширяет «Построение структурных моделей отказов».

4) Типовой сценарий выполнения: пользователь вводит исходные данные (с включенным вводом логики связей и агрегатов, при необходимости – загрузкой из файла), затем строит структурную модель (обязательные матрица смежности и схема сопряжения), после чего формирует структурные модели отказов (*RCA* и *FTA*) и при необходимости сохраняет результаты в файл.

В таблицах 6.8 – 6.10 представлено детальное описание диаграммы вариантов использования (*Use Case*).

Таблица 6.8 - Варианты использования верхнего уровня

Вариант использования	Назначение	Основные результаты	Задействованные подфункции
Ввод исходных данных	Задание исходной информации для дальнейшего моделирования	Загруженные/введенные данные; агрегаты; логика связей	Ввод логики связей ( <i>include</i> ); Ввод агрегатов ( <i>include</i> ); Загрузка данных из файла ( <i>extend</i> )
Построение структурной модели	Формирование структурного представления системы	Матрица смежности; схема сопряжения; при необходимости — меры важности; отображение схемы	Построение матрицы смежности ( <i>include</i> ); Построение схемы сопряжения ( <i>include</i> ); Расчет мер важности ( <i>extend</i> ); Вывод схемы сопряжения на экран ( <i>include</i> )
Построение структурных моделей отказов	Получение моделей причинно-следственных связей отказов	Модель <i>RCA</i> ; модель <i>FTA</i> ; по желанию - сохранение в файл	Построение <i>RCA</i> ( <i>include</i> ); Построение <i>FTA</i> ( <i>include</i> ); Сохранение результатов в файл ( <i>extend</i> )

Таблица 6.9 - Детализация подфункций

Базовый вариант	Отношение	Подфункция	Краткое описание результата
Ввод исходных данных	<i>include</i>	Ввод логики связей	Определение связей/отношений между сущностями данных
Ввод исходных данных	<i>include</i>	Ввод агрегатов	Задание агрегированных показателей для последующей обработки

Базовый вариант	Отношение	Подфункция	Краткое описание результата
Ввод исходных данных	extend	Загрузка данных из файла	Опциональная загрузка исходной информации из файла
Построение структурной модели	include	Построение матрицы смежности	Формирование матрицы смежности по введенной логике связей
Построение структурной модели	include	Построение схемы сопряжения	Получение визуальной/структурной схемы сопряжения
Построение структурной модели	extend	Расчет мер важности	Дополнительная оценка важности элементов/связей модели
Построение структурной модели	include	Вывод схемы сопряжения на экран	Представление полученной схемы пользователю
Построение структурных моделей отказов	include	Построение RCA	Построение модели Root Cause Analysis
Построение структурных моделей отказов	include	Построение FTA	Построение Fault Tree Analysis
Построение структурных моделей отказов	extend	Сохранение результатов в файл	Опциональная фиксация результатов на диске

Таблица 6.10 - Триггеры, входы и выходы

Вариант использования	Триггер	Входные данные	Выход/артефакт
Ввод исходных данных	Команда пользователя	Файл данных и/или ручной ввод; параметры агрегирования; описание связей	Набор исходных данных с агрегатами и логикой связей
Построение структурной модели	Команда пользователя	Подготовленные данные, логика связей	Матрица смежности; схема сопряжения; меры важности (при выполнении расширения); отображение схемы
Построение структурных моделей отказов	Команда пользователя	Структурная модель/данные об инцидентах	Модели RCA и FTA; файл с результатами (при выполнении расширения)

На рисунке 6.24 представлена Диаграмма активностей (*Activity Diagrams*).

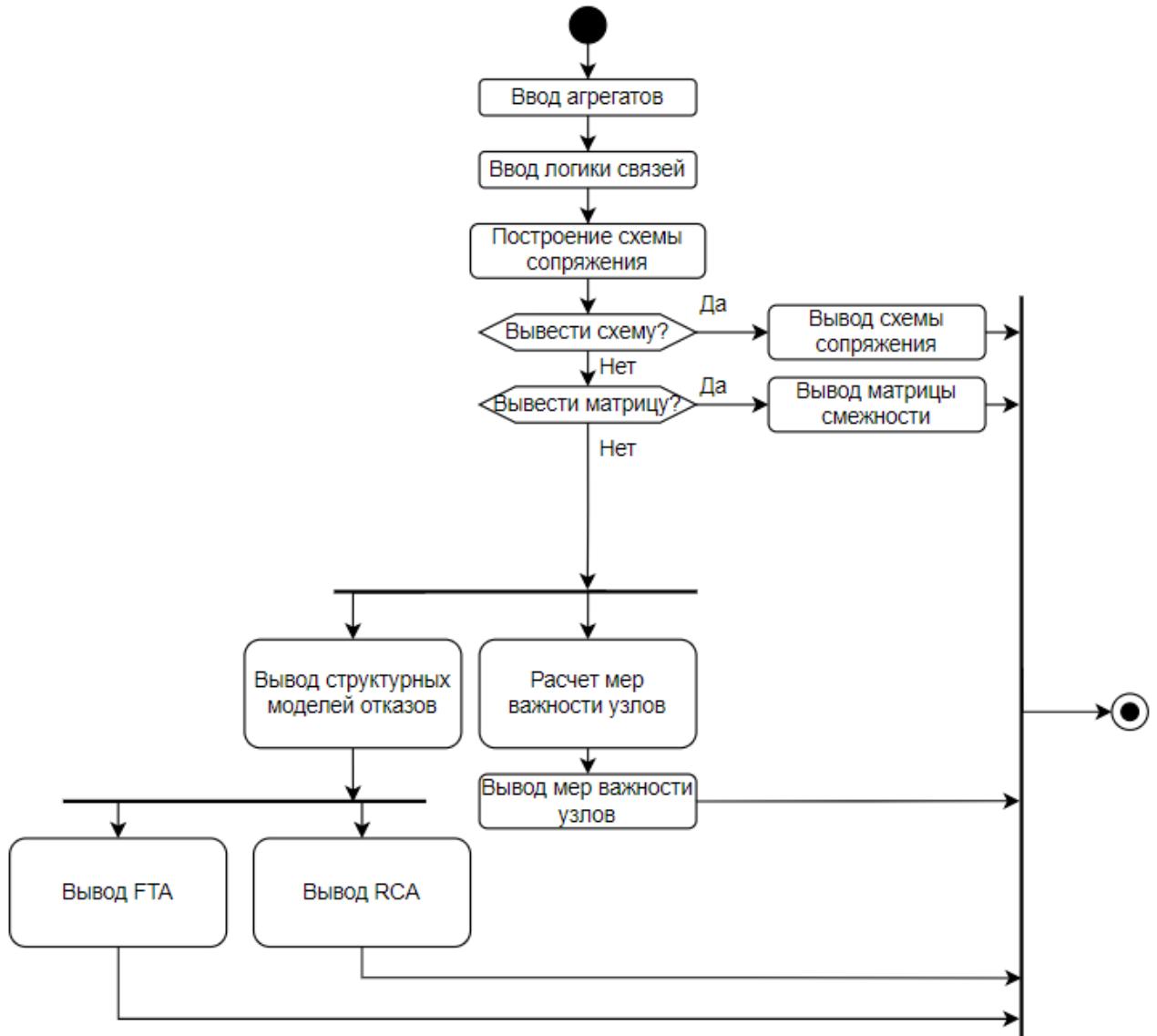


Рисунок 6.24 – Диаграмма активностей (*Activity Diagrams*)

Диаграмма активностей представляет собой графическое представление последовательности шагов, действий и решений. Используется для анализа требований и проектирования интерфейса, либо системы.

В таблицах 6.11 – 6.13 представлено детальное описание диаграммы активностей (*Activity Diagrams*).

Таблица 6.11 - Узлы диаграммы активностей

ID	Тип узла	Наименование	Назначение/ результат	Входы	Выходы
A0	Начало	Точка старта	Инициализация процесса	-	Поток к A1
A1	Действие	Ввод агрегатов	Задание агрегированных показателей для последующей обработки	Исходные данные пользователя	Поток к A2; артефакт Aggregates
A2	Действие	Ввод логики связей	Определение связей между элементами модели	Aggregates, исходные данные	Поток к A3; артефакт LinkRules
A3	Действие	Построение схемы сопряжения	Формирование структуры связей на основе LinkRules	Aggregates, LinkRules	Поток к D1; артефакт CouplingScheme
D1	Решение	Вывести схему?	Выбор визуализации схемы сопряжения	CouplingScheme	Да -> A4; Нет -> D2
A4	Действие	Вывод схемы сопряжения	Отображение CouplingScheme пользователю	CouplingScheme	Поток к F1
D2	Решение	Вывести матрицу?	Выбор табличного представления	CouplingScheme	Да -> A5; Нет -> F1
A5	Действие	Вывод матрицы смежности	Отображение матрицы смежности, полученной из CouplingScheme	CouplingScheme	Поток к F1; артефакт AdjacencyMatrix
D1	Решение	Вывести схему?	Выбор визуализации схемы сопряжения	CouplingScheme	Да -> A4; Нет -> D2
A4	Действие	Вывод схемы сопряжения	Отображение CouplingScheme пользователю	CouplingScheme	Поток к F1
D2	Решение	Вывести матрицу?	Выбор табличного представления	CouplingScheme	Да -> A5; Нет -> F1
A5	Действие	Вывод матрицы смежности	Отображение матрицы смежности, полученной из CouplingScheme	CouplingScheme	Поток к F1; артефакт AdjacencyMatrix
F1	Разветвление	Параллельное выполнение	Одновременный запуск ветвей анализа	CouplingScheme (и при наличии AdjacencyMatrix)	Потоки к A6 и A7

ID	Тип узла	Наименование	Назначение/ результат	Входы	Выходы
A6	Действие	Вывод структурных моделей отказов	Подготовка представления моделей отказов	CouplingScheme	Поток к J1; артефакты для FTA и RCA
A7	Действие	Расчет мер важности узлов	Вычисление показателей важности элементов модели	CouplingScheme, AdjacencyMatrix (при наличии)	Поток к A8; артефакт ImportanceMeasures
A8	Действие	Вывод мер важности узлов	Отображение ImportanceMeasures пользователю	ImportanceMeasures	Поток к J1
J1	Синхронизация	Сведение ветвей	Ожидание завершения A6 и A8	Потоки от A6 и A8	Потоки к A9 и A10 (допускается последовательный вывод)
A9	Действие	Вывод FTA	Представление Fault Tree Analysis	Данные из A6	Поток к A11
A10	Действие	Вывод RCA	Представление Root Cause Analysis	Данные из A6	Поток к A11
A11	Окончание	Точка завершения	Завершение процесса визуализации и расчетов	Потоки от A9 и A10	-

Таблица 6.12 - Переходы и условия

Откуда -> Куда	Условие/метка	Комментарий
A0 -> A1	-	Запуск процесса
A1 -> A2	-	После задания агрегатов вводится логика связей
A2 -> A3	-	На основе логики строится схема сопряжения
A3 -> D1	-	Переход к выбору формы вывода
D1 -> A4	Да	Отобразить CouplingScheme
D1 -> D2	Нет	Пропустить схему, перейти к вопросу о матрице
D2 -> A5	Да	Отобразить матрицу смежности
D2 -> F1	Нет	Переход к дальнейшим шагам без визуализации
A4 -> F1	-	После вывода схемы запускаются параллельные ветви
A5 -> F1	-	После вывода матрицы запускаются параллельные ветви
F1 -> A6	-	Ветвь 1: модели отказов
F1 -> A7	-	Ветвь 2: меры важности
A6 -> J1	-	Готовность данных для FTA/RCA
A7 -> A8	-	Публикация результатов расчета важности
A8 -> J1	-	Объединение ветвей
J1 -> A9	-	Вывод FTA
J1 -> A10	-	Вывод RCA
A9 -> A11	-	Завершение после FTA (если вывод RCA выполнен ранее)
A10 -> A11	-	Завершение после RCA (если вывод FTA выполнен ранее)

Таблица 6.13 - Артефакты данных

Артефакт	Источник	Используется в	Описание
Aggregates	A1	A2, A3	Агрегированные показатели для построения модели
LinkRules	A2	A3	Правила связи элементов модели
CouplingScheme	A3	D1, D2, A4, F1, A6, A7	Структурная схема сопряжения элементов
AdjacencyMatrix	A5 (или вычисляется)	A7	Табличное представление связности
ImportanceMeasures	A7	A8	Набор показателей важности узлов модели
FTA Model	A6	A9	Данные для Fault Tree Analysis
RCA Model	A6	A10	Данные для Root Cause Analysis

Пояснения к диаграмме активностей (рисунок 6.24):

- 1) Поток управления начинается с ввода агрегатов и логики связей. Эти два шага формируют основу для построения структуры системы, представленной как схема сопряжения.
- 2) После построения схемы пользователю предоставляются альтернативы визуализации: вывести схему, вывести матрицу смежности или пропустить оба шага. Любая из выбранных опций не влияет на дальнейшие вычисления, так как CouplingScheme уже сформирована.
- 3) Горизонтальная черта F1 обозначает параллельный запуск двух направлений: подготовка структурных моделей отказов и расчет мер важности узлов. Оба направления независимы и могут выполняться одновременно.
- 4) Результаты параллельных ветвей сводятся в J1. После синхронизации выполняется представление FTA и RCA. Порядок вывода FTA и RCA не критичен, допускается параллельная или последовательная публикация.
- 5) Точка завершения помещена после вывода FTA и RCA. Таким образом, сценарий завершается, когда результаты визуализированы и при необходимости показатели мер важности узлов выведены пользователю.
- 6) Диаграмма допускает конфигурации пользовательского интерфейса, в которых визуализация CouplingScheme и AdjacencyMatrix является optionalной, а ключевые аналитические шаги (вывод структурных моделей отказов, расчет и вывод мер важности узлов) выполняются всегда после построения схемы сопряжения.

На рисунке 6.25 представлена Диаграмма последовательностей (*Sequence Diagrams*). Диаграмма последовательностей в *UML* используется для визуализации взаимодействия между объектами или компонентами системы в рамках конкретного сценария.

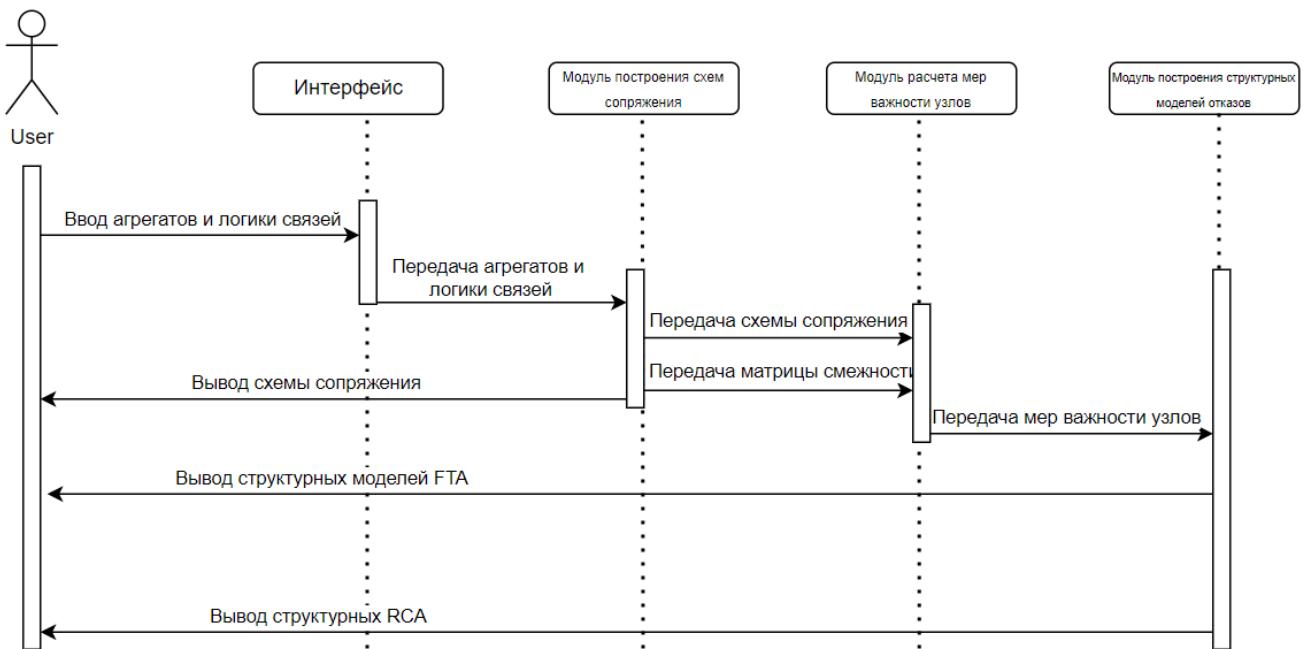


Рисунок 6.25 – Диаграмма последовательностей (*Sequence Diagrams*)

В таблицах 6.14 – 6.17 представлено детальное описание диаграммы последовательностей (*Sequence Diagrams*).

Таблица 6.14 - Участники взаимодействия

Обозначение	Роль	Краткое описание ответственности
User	Актор	Инициирует сценарий, вводит исходные данные и получает результаты вычислений и визуализаций
Интерфейс	Граничный компонент	Принимает ввод от User, валидирует и передает данные в вычислительные модули; возвращает пользователю результаты
Модуль построения схем сопряжения	Сервисный компонент	Формирует структурное представление системы: схему сопряжения и матрицу смежности
Модуль расчета мер важности узлов	Сервисный компонент	Вычисляет меры важности узлов на основе схемы сопряжения и/или матрицы смежности
Модуль построения структурных моделей отказов	Сервисный компонент	Формирует структурные модели отказов (FTA, RCA), используя входные данные и, при наличии, меры важности узлов

Таблица 6.15 – Сообщения и порядок обмена

№	Отправитель -> Получатель	Наименование сообщения	Содержимое/ данные	Тип вызова	Ожидаемый результат
S1	User -> Интерфейс	Ввод агрегатов и логики связей	Агрегаты; правила/логика связей	Синхронный	Интерфейс фиксирует ввод и готовит передачу в сервис
S2	Интерфейс -> Модуль построения схем сопряжения	Передача агрегатов и логики связей	Агрегаты; логика связей	Синхронный	Модуль начинает построение структуры
S3	Модуль построения схем сопряжения -> Интерфейс	Передача схемы сопряжения	Структура/граф связей	Синхронный	Интерфейс получает схему для возможной визуализации
S4	Модуль построения схем сопряжения -> Интерфейс	Передача матрицы смежности	Матрица смежности	Синхронный	Интерфейс получает табличное представление связей.
S5	Интерфейс -> User	Вывод схемы сопряжения	Визуализация схемы	Асинхронный/по событию	Пользователь видит схему сопряжения.
S6	Модуль расчета мер важности узлов <- Модуль построения схем сопряжения	Передача входных структур	Схема сопряжения и/или матрица смежности	Синхронный	Запуск расчета мер важности узлов.
S7	Модуль построения структурных моделей отказов <- Модуль расчета мер важности узлов	Передача мер важности узлов	Набор метрик/рангов узлов	Синхронный	Модуль отказов дополняет модели приоритизацией
S8	User <- Модуль построения структурных моделей отказов (через Интерфейс)	Вывод структурных моделей FTA	Визуализация/презентация FTA	Асинхронный/по готовности	Пользователь получает модель FTA
S9	User <- Модуль построения структурных моделей отказов	Вывод структурных RCA	Визуализация/презентация RCA	Асинхронный/по готовности	Пользователь получает модель RCA

**Примечания:**

- сообщения S3 и S4 могут приходить последовательно; визуализация (S5) инициируется интерфейсом по политике UI;
- ветви построения мер важности (S6-S7) и формирования моделей отказов/их вывода (S8-S9) допускают параллельное исполнение после получения структуры (S3/S4).

**Таблица 6.16 – Данные и артефакты**

Артефакт	Источник	Потребители	Назначение
Агрегаты	User через Интерфейс	Модуль построения схем сопряжения	Сводные показатели для построения структуры
Логика связей	User через Интерфейс	Модуль построения схем сопряжения	Правила построения отношений между элементами
Схема сопряжения	Модуль построения схем сопряжения	Интерфейс; Модуль расчета мер важности узлов	Графовое представление структуры системы
Матрица смежности	Модуль построения схем сопряжения	Интерфейс; Модуль расчета мер важности узлов	Табличное представление связности
Меры важности узлов	Модуль расчета мер важности узлов	Модуль построения структурных моделей отказов	Приоритизация узлов для анализа отказов
Модель FTA	Модуль построения структурных моделей отказов	User (через Интерфейс)	Дерево отказов для визуализации/интерпретации
Модель RCA	Модуль построения структурных моделей отказов	User (через Интерфейс)	Представление Root Cause Analysis

**Таблица 6.17 - Предусловия и постусловия**

Этап	Предусловия	Постусловия
Ввод (S1)	Пользователь обладает исходными агрегатами и правилами связей	Данные валидированы интерфейсом и готовы к передаче (S2)
Построение структуры (S2 – S4)	Доступны агрегаты и логика связей	Сформированы схема сопряжения и матрица смежности
Расчет важности (S6 – S7)	Получены схема/матрица	Сформирован набор мер важности узлов
Формирование моделей отказов (S8 – S9)	Доступны структура и, при наличии, меры важности	Пользователь получил визуализации FTA и RCA

Пояснения к диаграмме последовательностей (рисунок 6.25):

- 1) Сценарий запускается пользователем через интерфейс, где вводятся агрегаты и логика связей. Интерфейс передает эти данные в модуль построения

схем сопряжения, который формирует два представления структуры: схему сопряжения и матрицу смежности. Интерфейс может сразу визуализировать схему для пользователя, не блокируя дальнейшие расчеты.

2) После получения структуры модуль расчета мер важности узлов выполняет вычисления на основе схемы и/или матрицы. Результат - набор метрик важности передается в модуль построения структурных моделей отказов, который использует их для формирования элементов в моделях *FTA* и *RCA*. По готовности результатов интерфейс предоставляет пользователю визуализации *FTA* и *RCA*. Порядок вывода этих моделей не является критичным и определяется настройками интерфейса.

На рисунке 6.26 представлена диаграмма компонентов (*Component Diagram*). Диаграмма компонентов помогает проектировать архитектуру программного обеспечения, показывает, из каких компонентов (логических или физических) состоит система и как они связаны. Основное достоинство данной диаграммы - упрощает понимание модульной структуры системы.

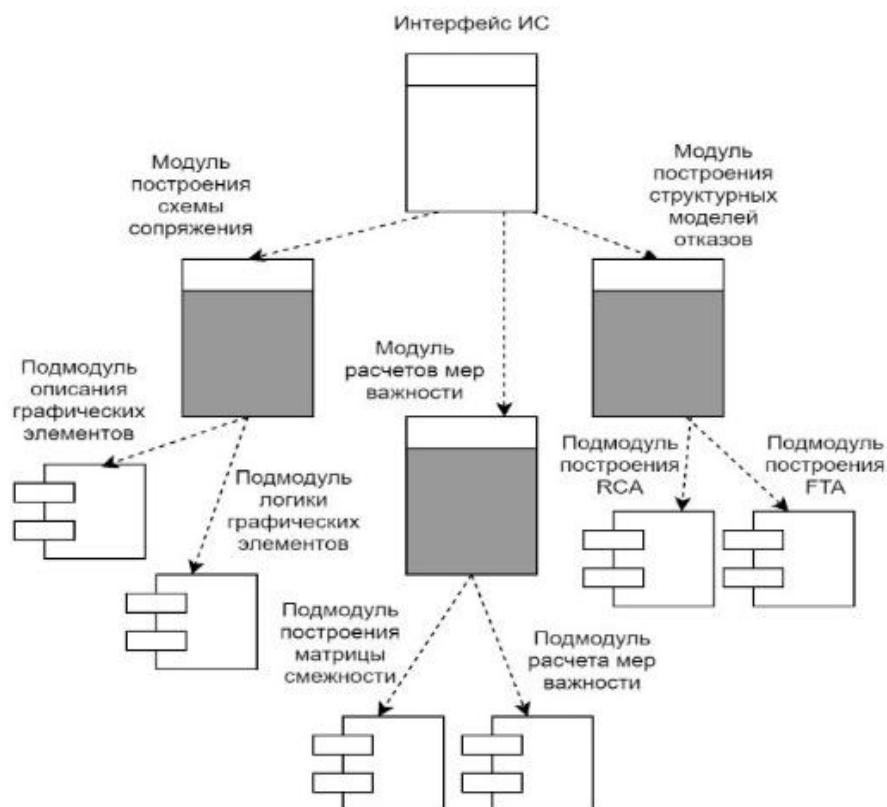


Рисунок 6.26 – Диаграмма компонентов

В таблицах 6.18 – 6.20 представлено детальное описание диаграммы компонентов.

Таблица 6.18 - Подмодули и их функции

Родительский модуль	Подмодуль	Функция/результат
Модуль построения схем сопряжения	Подмодуль описания графических элементов	Определение типов узлов/связей и их атрибутов для представления структуры
	Подмодуль логики графических элементов	Правила и ограничения построения/трансформации структуры (допустимые соединения, фильтры)
	Подмодуль построения матрицы смежности	Генерация матричного представления связности по схеме сопряжения
Модуль расчетов мер важности	Подмодуль расчета мер важности	Реализация алгоритмов вычисления показателей важности (узлы, ребра, подсети)
Модуль построения структурных моделей отказов	Подмодуль построения FTA	Построение деревьев отказов (Fault Tree Analysis)
	Подмодуль построения RCA	Конструирование моделей причинно-следственного анализа (RCA)

Таблица 6.19 - Верхнеуровневые компоненты и их ответственность

Компонент	Назначение	Входы	Выходы	Зависимости/взаимодействия
Интерфейс	Точка взаимодействия с пользователем; сбор и валидация входных данных; инициирование вычислительных сценариев; представление результатов	Параметры, агрегаты, логика связей	Визуализации и отчеты (схема сопряжения, матрица смежности, меры важности, модели FTA/RCA)	Вызывает модули: построения схем сопряжения; расчетов мер важности; построения структурных моделей отказов
Модуль построения схем сопряжения	Формирует структурное представление системы	Агрегаты; логика связей	Схема сопряжения (граф); матрица смежности	Использует подмодули: описания графических элементов; логики графических элементов; построения матрицы смежности
Модуль расчетов мер важности	Вычисляет количественные показатели важности элементов/связей структуры	Схема сопряжения; матрица смежности	Меры важности узлов/ребер	Использует подмодуль расчета мер важности; результаты передает в модуль построения структурных моделей отказов и в интерфейс
Модуль построения структурных моделей отказов	Формирует модели анализа отказов	Схема сопряжения; (опционально) меры важности	Структурные модели FTA и RCA	Использует подмодули построения FTA и RCA; результаты возвращает в интерфейс

Таблица 6.20 Границы ответственности и точки расширения

Область	Ответственность	Возможные расширения
Представление/UX	Интерфейс ИС	Дополнительные режимы визуализации схем/матриц/моделей; экспорт отчетов
Построение структуры	Модуль построения схем сопряжения	Новые типы узлов/ребер и правила логики; альтернативные способы генерации матрицы смежности
Метрики важности	Модуль расчетов мер важности	Подключение дополнительных алгоритмов оценки важности; поддержка весов/временной динамики
Анализ отказов	Модуль построения структурных моделей отказов	Расширение библиотек шаблонов <i>FTA</i> ; добавление методов <i>RCA</i> (варианты причинно-следственных графов)

Пояснения к диаграмме компонентов (рисунок 6.26):

- 1) Композиция и слои. Схема разделяет систему на три вычислительных домена и один граничный слой. Граничный слой (Интерфейс) изолирован от алгоритмической части: не содержит доменной логики, а лишь собирает входные данные, инициирует операции и отображает результаты.
- 2) Формирование структуры. Модуль построения схем сопряжения - исходная точка анализа: на основе агрегатов и логики связей формирует графовую модель (схему сопряжения) и ее матрицу смежности. Для этого используются подмодули описания и логики графических элементов (тиปизация узлов/ребер, ограничения соединений) и подмодуль построения матрицы смежности.
- 3) Количественная оценка. Модуль расчетов мер важности получает схему сопряжения и/или матрицу смежности и вычисляет показатели мер важности узлов. Результаты возвращаются в интерфейс для визуализации и одновременно используются как вход для модуля анализа отказов.
- 4) Моделирование отказов. Модуль построения структурных моделей отказов строит: *FTA* и *RCA*. Подмодули *FTA* и *RCA* независимы.
- 5) Потоки и зависимости. Основные потоки направлены сверху вниз: интерфейс инициирует действия, далее данные движутся от построения структуры к метрикам важности и моделям отказов. Обратные потоки — это возврат артефактов для визуализации. Пунктирные связи на схеме трактуются как зависимость/вызов сервиса, сплошные - как передача данных.

6) Согласованность артефактов. Все вычислительные модули опираются на единые артефакты: граф (схема сопряжения), матрица смежности, меры важности узлов, модели *FTA/RCA*. Это снижает связанность, облегчает тестирование и обеспечивает возможность замены/расширения алгоритмов без изменений интерфейса.

Таким образом, комплекс разработанных объектно-ориентированных моделей является основой для разработки прототипа программного обеспечения для анализа моделей отказов функциональных компонент ИС по критериям функциональной надежности.

Прототип программы является практической реализацией системы, спроектированной на основе комплекса объектно-ориентированных моделей (*UML*-диаграмм), реализован на языке *Python*.

Использование *Python* объясняется: синтаксической простотой, богатым набором библиотек и поддержкой объектно-ориентированного подхода для реализации сложных систем.

Разработанный прототип программы представляет собой визуальный редактор для моделирования отказов. Прототип имеет модульную архитектуру, система разбита на независимые компоненты, что позволяет в дальнейшем упростить интеграцию с внешними сервисами.

Пользователь может создавать элементы системы (входы, агрегаты, выходы), соединять их, моделировать отказы и анализировать поведение системы с помощью *FTA* и *RCA*.

Программа обеспечивает наглядность представления информации, что облегчает интерпретацию результатов экспериментов.

Интерфейс реализован на *Tkinter* и поддерживает манипуляции с объектами, взаимодействие через контекстное меню и визуализацию структуры связей. В таблице 6.21 представлены основные функции, реализованные в программе.

Таблица 6.21 – Основной набор функций

Функциональность:	Набор функций:
Построение схемы сопряжения	add_input_node() – добавить входной узел; add_output_node() – добавить выходной узел; add_aggregate() – добавить агрегат; draw_node(node) – отрисовка узла; draw_connection(start, end, ...) – отрисовка соединения;
Взаимодействие пользователем	canvas_click(event) – обработка кликов по холсту; drag_start(event), drag(event), drag_stop(event) – перемещение узлов; set_delete_mode(), handle_delete_click(event) – режим удаления; start_connection(point), create_new_connection(...), stop_connection() – соединение точек;
Работа с соединениями	get_node_by_point(point_id) – поиск узла по точке; update_connections(node) – обновление соединений;
Анализ и визуализация	show_adjacency_matrix() – отобразить матрицу смежности; build_fault_tree() – построить FTA; build_rca_tree() – построить RCA;
Моделирование отказов	set_failure() — активация режима отказа; mark_failed_elements(point_id) — размечает отказ; color_failed_elements() — визуально отображает отказы; reset_failures() — сбросить отказы;
Прочее	reset_canvas() — сброс схемы; exit_modes(event) — выйти из всех режимов; on_closing() — завершение работы приложения; run() — запуск GUI.

Разработанный прототип позволяет строить схемы сопряжения из элементов, представленных на рисунке 6.27.

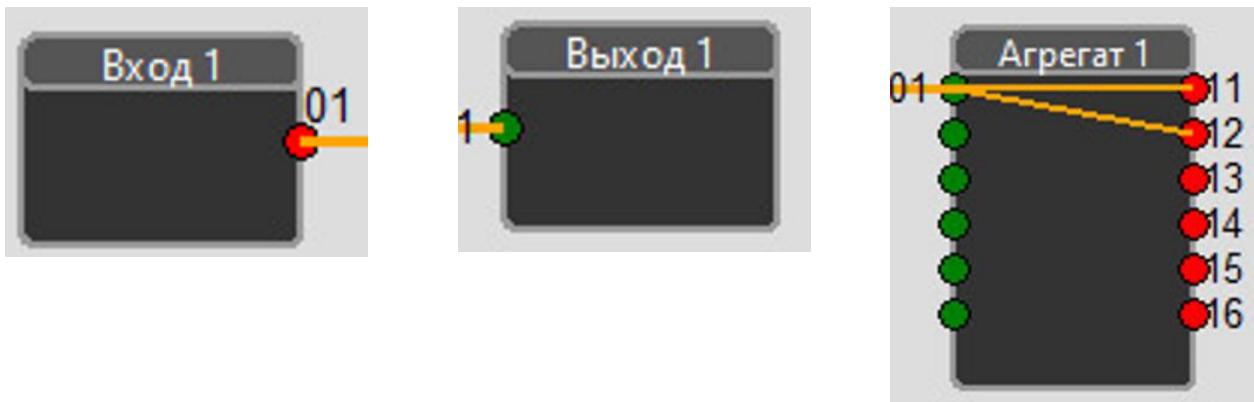


Рисунок 6.27 – Элементы для построения схемы сопряжения

Данные элементы (рисунок 6.27) перетаскиваются в рабочую область с помощью мыши, пример построенной схемы сопряжения представлен на рисунке 6.28.

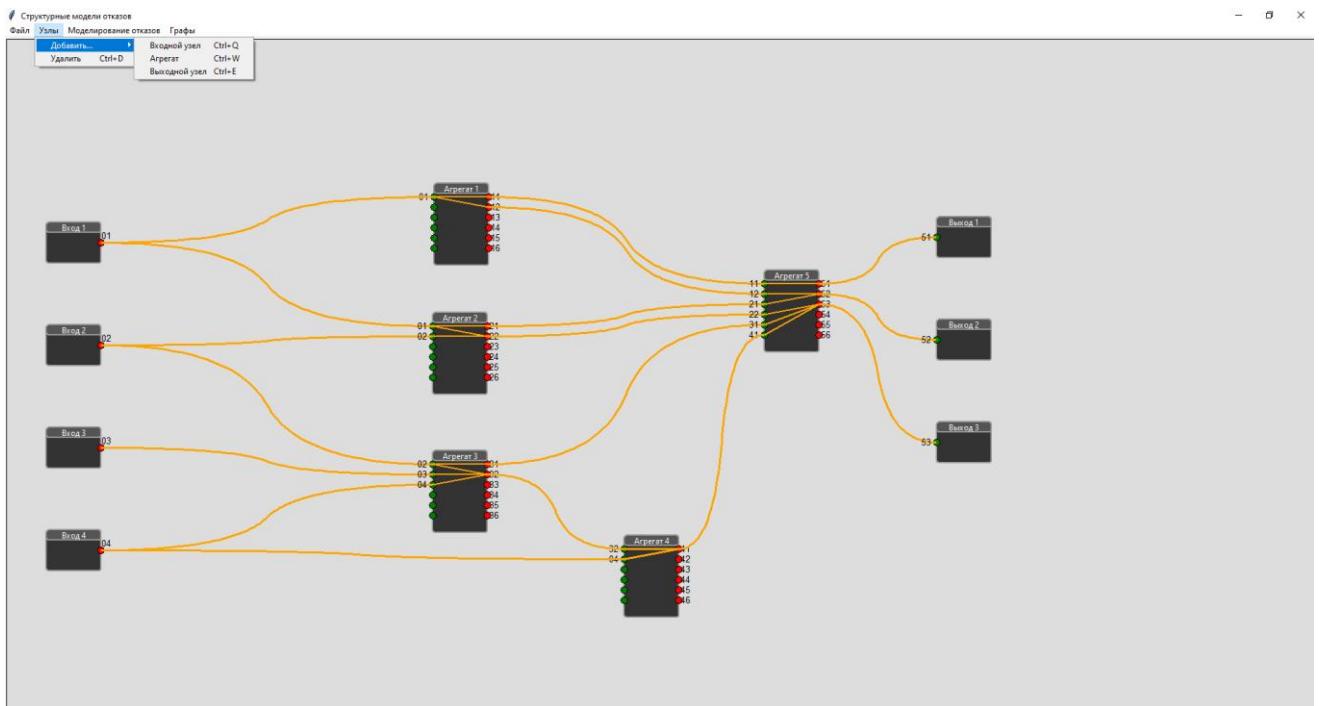


Рисунок 6.28 – Пример построенной схемы сопряжения

Схема сопряжения преобразуется в граф и строится матрица смежности, где единицам соответствуют светлые (зеленые) клетки. Пример матрицы для схемы, представленной на рисунке 6.28, представлен на рисунке 6.29.

Матрица смежности														
	01	02	03	04	11	12	21	22	31	32	41	51	52	53
01														
02														
03														
04														
11														
12														
21														
22														
31														
32														
41														
51														
52														
53														

Рисунок 6.29 – Пример матрицы смежности

На основе матрицы смежности рассчитываются меры важности узлов.

Пример расчета представлен на рисунке 6.30.

	I1	I2	degree centrality	page rank
01	2	6	3	0,026906
02	3	8	2	0,026906
03	1	3	2	0,026906
11	2	2	3	0,035874
12	1	2	2	0,035874
21	1	2	2	0,035874
22	3	6	4	0,040359
23	1	2	3	0,053812
31	1	1	3	0,098655
32	1	1	2	0,040359
41	1	1	2	0,040359
42	1	1	4	0,107623
51	0	0	1	0,044843
52	0	0	3	0,183857
53	0	0	1	0,067265
54	0	0	1	0,134529

Рисунок 6.30 – Пример результатов расчета мер важности узлов графа

Для того, чтобы визуально оценить распространение отказа вследствие отказа одной из вершин, в программе предусмотрена функция «подсвечивания» красными линиями всех путей, по которым произойдет распространение, пример приведен на рисунке 6.31.

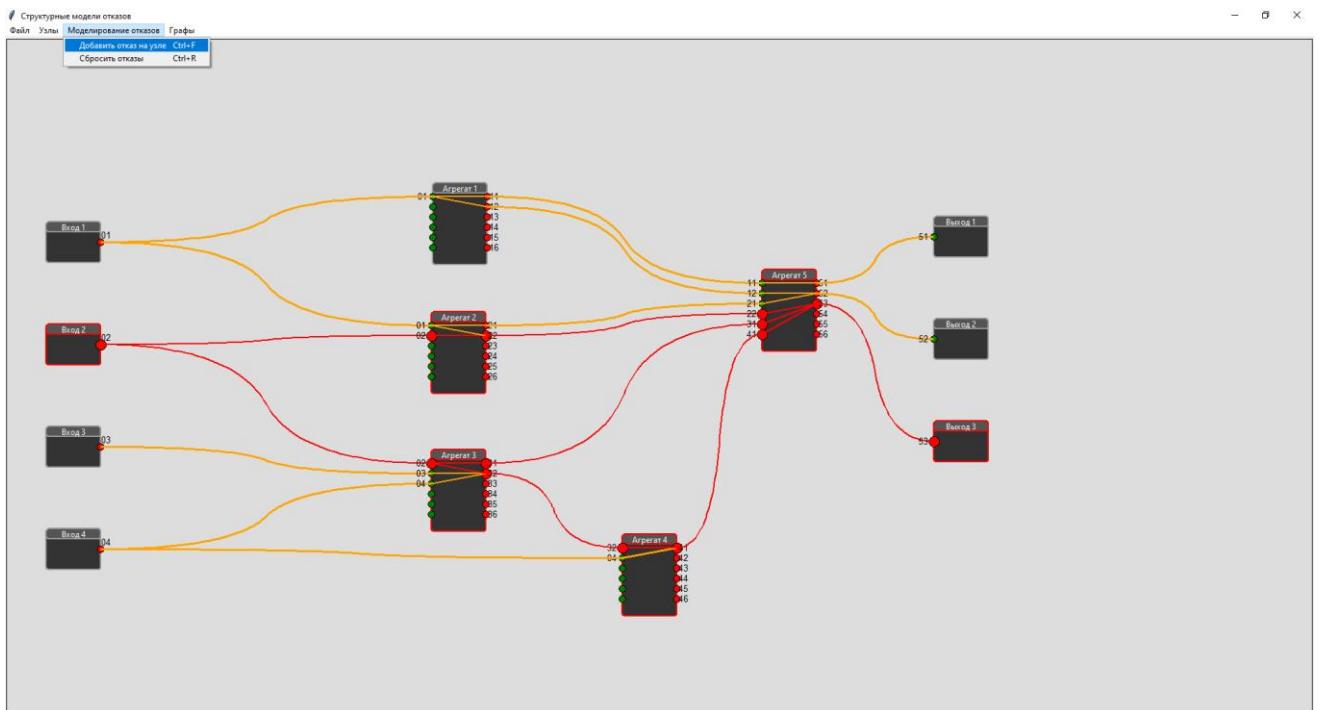


Рисунок 6.31 – Визуализация распространения ошибки (ошибка в узле 02)

В программе предусмотрена возможность автоматического построения на основе схемы сопряжения структурных моделей отказов. На рисунке 6.32 представлен пример построения дерева отказов (*FTA*), а на рисунке 6.33 анализ коренных причин (*RCA*).

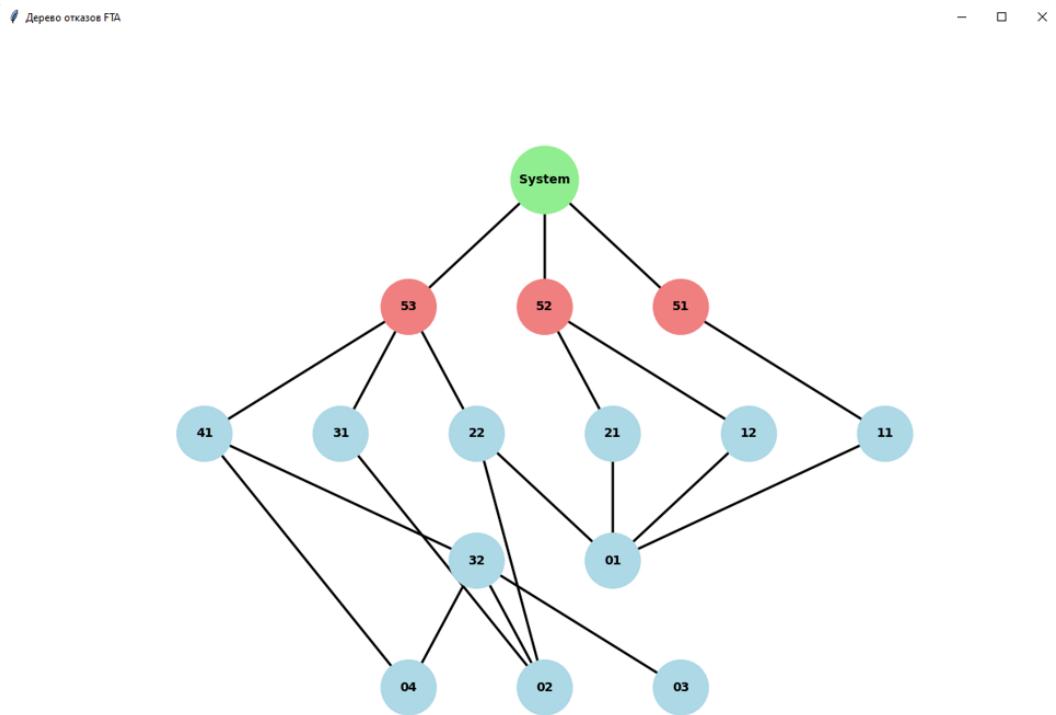


Рисунок 6.32 – Пример автоматического построения *FTA*

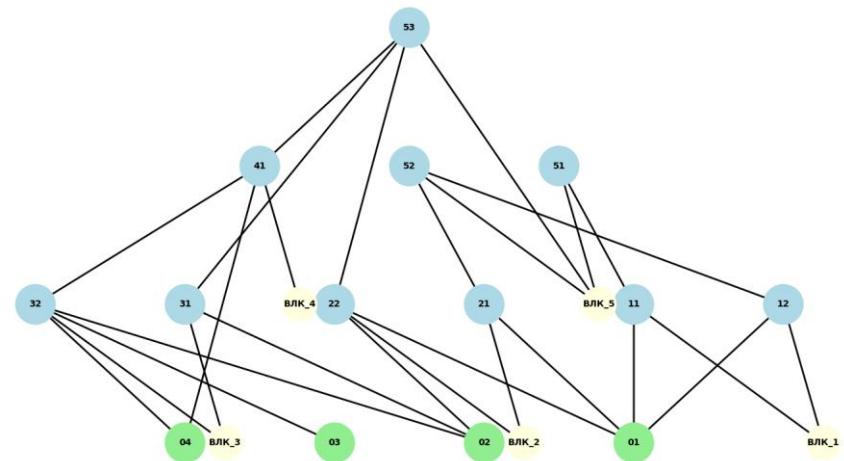


Рисунок 6.33 – Пример автоматического построения *RCA*

Разработанный прототип позволяет строить и анализировать схемы сопряжения агрегатов. В настоящее время в силу сложности и масштабности разрабатываемых программных компонентов, становится трудоемким процесс разработки структурных моделей отказов. Для решения этой проблемы в предлагаемом прототипе предусмотрена функция автоматического построения *FTA* и *RCA* по схеме сопряжения.

Наличие структурных моделей отказов позволяет определить наиболее важные с точки зрения функциональной надежности компоненты ИС и обоснованно распределить ресурсы.

Разработанный прототип может представлять ценность для исследователей, проектировщиков, разработчиков и тестировщиков ИС, а также использоваться в рамках методики для оценки возможных последствий отказов функциональных компонент ИС.

Методика анализа моделей проектируемых компонентов ИС по критериям функциональной надежности:

*Этап 1.* Построение схемы сопряжения агрегатов и формирование таблиц истинности.

В общем случае система рассматривается как многоуровневая конструкция из взаимодействующих элементов, при необходимости объединяемых в разные подсистемы. Взаимодействие элементов в процессе функционирования системы рассматривается как результат совокупности воздействий между элементами посредством сигналов.

На данном этапе необходимо определить, как различные функциональные компоненты ИС будут взаимодействовать между собой, определить вход, выход для каждого агрегата, т.е. построить схему сопряжения агрегатов (пример схемы сопряжения агрегатов с помощью разработанного прототипа представлен на рисунке 6.28), далее построенная схема преобразуется в граф и строится матрица смежности (пример матрицы смежности представлен на рисунке 6.28).

*Этап 2.* Расчет мер важности узлов на основе построенной схемы сопряжения агрегатов.

На данном этапе выполняется расчет мер важности узлов графа.

Показатель  $I1$  – соответствует числу выходов системы, которые будут задеты отказом рассматриваемой вершины (в эту сумму не входит сама вершина).

Показатель  $I2$  – общее число вершин графа, на которое распространится отказ в заданной вершине.

*PageRank* – степень влиятельности, рассчитывается аналогично *PageRank*, алгоритм предложенный *Google*, для ранжирования веб-страниц.

Все перечисленные показатели рассчитываются по матрице смежности или путем ее преобразования. Для расчета *PageRank* начальные значения  $I$  положены равными числу, обратному общему числу вершин графа (пример расчета мер важности узлов графа представлен на рисунке 6.30).

*Этап 3.* Построение на основе схемы сопряжения структурной модели отказов (*FTA*). Совместное использование схем сопряжения и таблиц истинности создает основу для построения структурных моделей.

На данном этапе в рамках нисходящего подхода строиться структурная модель отказов (*FTA*). Пример структурной модели *FTA*, построенной с помощью разработанного прототипа представлен на рисунке 6.32.

*Этап 4.* Выявление коренных причин отказов (*RCA*). На данном этапе в рамках нисходящего подхода строиться структурная модель выявления коренных причин отказов на основе подхода *RCA*.

Пример структурной модели *RCA*, построенной с помощью разработанного прототипа представлен на рисунке 6.33.

*Этап 5.* Анализ полученных результатов.

На данном этапе на основе полученных структурных моделей отказов определяются наиболее важные компоненты ИС, что позволяет обоснованно распределить ресурсы на обеспечение функциональной надежности. А также провести оценку последствий отказов на предпроектной стадии.

На рисунке 6.34 схематично представлены основные этапы методики анализа моделей проектируемых компонентов ИС.

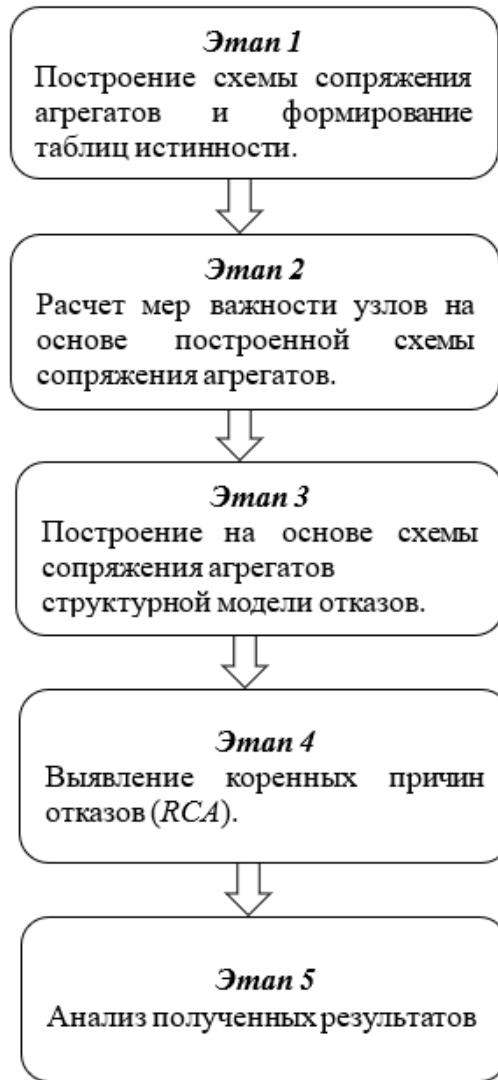


Рисунок 6.34 – Основные этапы методики анализа моделей проектируемых компонентов ИС

Разработанную методику для анализа моделей проектируемых компонентов ИС рекомендуется применять проектировщикам, разработчикам и тестировщикам ИС на предпроектной стадии.

Методика и прототип программного обеспечения для анализа моделей проектируемых компонентов ИС создают информационную основу для сравнения альтернативных вариантов проектных решений по критерию функциональной надежности на предпроектной стадии.

## Выводы по главе 6

1. В рамках реактивного подхода проанализированы проблемные ситуации, возникающие при работе с локальными ИС на основе поступивших запросов в службу поддержки, выявлены коренные причины, способствующие возникновению нежелательных событий. Использование коэффициента готовности в качестве интегрального показателя функциональной надежности позволяет объединить результаты *RCA* и *FTA* в единую метрику, отражающую фактическое состояние ИС. Благодаря интеграции реактивного и проактивного подхода достигается системный эффект, что способствует повышению функциональной надежности ИС.

2. В рамках барьерного подхода на основе полученных теоретических результатов предложена методика формальной оценки возможности реализации проекта, позволяющая оценить предполагаемую удовлетворенность заказчика результатами проекта, а исполнителей – ходом проекта в зависимости от бюджета и ограничений на длительность проекта с учетом опыта, полученного от ранее реализованных проектов, на на предпроектной стадии. Это позволяет получить экономический эффект за счет сокращения сроков предпроектной стадии и повышения обоснованности параметров проекта.

3. На основе предложенного метода структурного моделирования последствий отказов на основе аппарата схем сопряжения и таблиц истинности разработана методика и прототип программного обеспечения, позволяющего проанализировать модели проектируемых компонентов ИС и оценить альтернативные варианты проектных решений по критерию функциональной надежности на предпроектной стадии, а также избавиться от трудоемкого и затратного по времени процесса ручного построения структурных моделей отказов.

Основные результаты, представленные в шестой главе, подтверждаются актами внедрения (копии актов представлены в приложении А), свидетельством об официальной регистрации программы для ЭВМ [102 - 106], опубликованы в рецензируемых печатных изданиях из перечня, утвержденного ВАК [21, 22].

## ОСНОВНЫЕ РЕЗУЛЬТАТЫ И ВЫВОДЫ

В результате диссертационного исследования разработаны теоретические, методологические и модельные основы обеспечения функциональной надежности ИС на предпроектной стадии. При этом сделаны следующие выводы и получены результаты:

1. Разработана методология обеспечения функциональной надежности ИС на предпроектной стадии, основанная на интеграции системного, архитектурного, динамического и других общенаучных подходов и принципов. Предложенная методология отличается тем, что она объединяет в себе реактивный и проактивный подходы к обеспечению функциональной надежности ИС на предпроектной стадии. В рамках системного подхода разработаны концептуальные основы обеспечения функциональной надежности, отличающиеся от известных подходов, тем, что с единых позиций позволяют рассмотреть стратегии, реализующие реактивный и проактивный подходы. Впервые сформулированы принципы и подходы к обеспечению функциональной надежности ИС как разновидности сложных систем. В совокупности это позволило сформировать методологическую основу для исследования структуры проблемных ситуаций, связанных с обеспечением функциональной надежности ИС на предпроектной стадии.

2. На основе положений эвергетики и системной интеграции *V*-модели жизненного цикла и подхода *QFD* (функции развертывания качества), реализованного в виде модели «Дом качества» - *NoQ*, разработаны новые системные модели предпроектной стадии. Разработанный комплекс системных моделей предпроектной стадии отличается тем, что является основой для выработки консолидированного мнения неоднородных акторов относительно внешнего облика ИС, а также позволяет в единой форме представить преобразования дефектов при переходах на разные стадии жизненного цикла. Комплексное использование системных моделей позволяет повысить прозрачность содержания ожидаемых результатов проекта для заинтересованных сторон.

3. Разработаны методы обеспечения функциональной надежности ИС на предпроектной стадии, включающие:

- метод построения многомерных регрессионных зависимостей отличающийся от известных тем, что исходные данные, соответствующие разным компонентам векторов независимой и зависимой переменных, представляются либо в виде результатов измерений, либо в виде разных по форме экспертных оценок. Основу метода составляет преобразование исходных данных, изначально представляемых в разной форме, к виду закона распределения непрерывной случайной величины.

- метод оценки функциональной надежности информационных сервисов с учетом субъективных оценок пользователей, основанный на концепции профилей. Основу метода составляет преобразование различных по форме исходных данных к единому виду. Данный подход отличается от существующих тем, что позволяет получать метрические оценки надежности информационных сервисов на основе совокупного использования как измерительных данных, так и субъективных оценок пользователей. Это делает возможным обоснованное определение направления повышения функциональной надежности ИС.

- метод структурного моделирования последствий отказов функциональных компонент ИС, отличающийся от известных методов совместным использованием аппарата схем сопряжения и таблиц истинности. Это создает основу для построения структурных моделей, обеспечивающих моделирование отказов в рамках как нисходящего, так и восходящего подходов к исследованию. Метод позволяет провести оценку возможных последствий отказов по различным критериям функциональной надежности для разных схем построения ИС на предпроектной стадии.

4. Разработаны структурные модели процессов обеспечения функциональной надежности ИС, основанные на системном сочетании архитектурного подхода и анализе структурных свойств, отличаются тем, что пространство проблемных ситуаций, связанных с обеспечением функциональной надежности, формируется на основе системных архетипов.

Разработана формальная модель ошибок предпроектной стадии и выполнена ее верификация. Установлено, что предложенной модели может быть поставлен в соответствие архетип «Пределы роста», из чего следует, что существует предел возможностей по предупреждению дефектов. Предлагаемый подход отличается тем, что источники ошибок и дефектов выявляются на предпроектной стадии, в отличие от известных подходов, которые ориентируются на то, что проект начинается с разработки и реализации технического задания.

Разработаны контурные и динамические модели проблемных ситуаций, связанных с обеспечением функциональной надежности ИС на основе системных архетипов, отличаются тем, что системные архетипы представляются совокупностью контурных моделей, весовые характеристики которых определяются математико-статистическими методами, это позволяет перейти от качественного описания ситуаций к математическим моделям, позволяющим получить количественные оценки отдельных ситуаций.

В рамках ресурсного подхода предложена модель формирования интегрального показателя, характеризующего совокупное влияние факторов, как способствующих, так и препятствующих повышению функциональной надежности, отличается тем, что представляет собой системное сочетание модели «Дом качества» и системных архетипов. Это позволяет повысить степень формализации исследований, ориентированных на оценку влияния качества управления проектом на функциональную надежность ИС.

5. Предложена статическая модель проекта как многосвязного объекта управления основанная на том, что состояние проекта характеризуется совокупностью прямых и перекрестных связей между входными и выходными параметрами, отличается тем, что характеристики прямых и перекрестных связей определяются на основе исторических данных о ранее реализованных проектах, представленных в виде числовых характеристик, и субъективных оценок лиц, причастных к реализации прежних проектов. Предложенная формальная модель дает возможность повысить обоснованность принятия стратегических решений по организации проекта основными заинтересованными сторонами.

Предложен подход к оцениванию согласованности точек зрения неоднородных акторов на основе их субъективного восприятия значимости проблемных ситуаций, отличается тем, что для основы формализации описания видений неоднородных акторов объединяет в себе системные архетипы и модель «Дом качества» - HoQ. В качестве индикаторов изменения степени согласованности точек зрения выступают парциальное и множественный коэффициенты корреляции. Предложенная схема формирования метрических характеристик степени согласованности мнений неоднородных акторов создает основу для мониторинга хода выработки консолидированного мнения неоднородными акторами о внешнем облике ИС, что сокращает длительность предпроектной стадии.

6. На основе предложенных принципов и подходов к обеспечению функциональной надежности выявлены факторы функциональной надежности ИС при формировании единого информационного пространства компании.

В рамках барьерного подхода к предотвращению организационных дефектов предложена методика формальной оценки возможности реализации проекта, отличается тем, что позволяет на предпроектной стадии оценить предполагаемую удовлетворенность заказчика результатами проекта, а исполнителей – ходом проекта в зависимости от бюджета и ограничений на длительность проекта с учетом опыта, полученного от ранее реализованных проектов. Это позволяет получить экономический эффект за счет сокращения сроков предпроектной стадии и повышения обоснованности параметров проекта.

На основе предложенного метода структурного моделирования последствий отказов на основе аппарата схем сопряжения и таблиц истинности разработана методика и программное обеспечение, отличается тем, что позволяет строить структурные модели в рамках как нисходящего (дерево неисправностей FTA, модели причинно-следственных связей Fishbone), так и восходящего подходов к исследованию функциональной надежности (дерево событий ETA) и избавиться от трудоемкого и затратного по времени процесса ручного построения структурных моделей отказов.

## ГЛОССАРИЙ

**Информационная система (ИС)** - система, организующая обработку информации о предметной области и ее хранение [53].

Понятие ИС в широком смысле подразумевает, что её неотъемлемыми компонентами являются данные, техническое и программное обеспечение, а также персонал и организационное обеспечение.

**Функциональной надежность ИС** - свойство сохранять работоспособность в соответствии со своим целевым назначением при случайных дестабилизирующих воздействиях и отсутствии злоумышленного влияния на программную и аппаратную составляющую.

**Система сетецентрического управления** представляет собой сочетание заблаговременно созданных и развернутых разветвленных автоматизированных электронных сетей сбора и обработки первичной информации, узлов хранения и анализа информации, а также контуров управления и принятия решений. Их интеграция формирует единое информационное и управляемое пространство [68, 69].

**Интероперабельность** (англ. *Interoperability*) - ключевое свойство сетецентрических систем, обеспечивающее их способность к совместному функционированию и обмену данными между разнородными компонентами, подсистемами и участниками в рамках распределённой информационной среды [65, 79, 128].

**Киберфизические системы (Cyber-Physical System, CPS)** – это системы, состоящие из различных природных объектов, искусственных подсистем и управляющих контроллеров, позволяющих представить такое образование как единое целое. Киберфизические системы являются системной интеграцией физического и виртуальных миров [68, 117].

**Проблемная ситуация** – ситуация невозможности достичь какой-либо насущной цели с использованием наличных ресурсов и с учётом существующих ограничений (временных, материальных, организационных, интеллектуальных и других).

**Системный архетип** (от греч. *arche* – начало и *typos* – образ) трактуется как «начальный образ», формальный шаблон, отражающий устойчивую структуру проблемы, идея. В [84, 92, 97, 98, 107] подчёркивается значимость архетипов как инструмента, позволяющего структурировать типовые по содержанию управленические проблемы, характерные для функционирования сложных систем.

**Эвергетика** - субъектно- и ценностно-ориентированная наука о процессах управления в обществе, связанную в большей степени с решением плохо формализуемых проблем управления. Важным для эвергетики направлением исследований является организация взаимодействия акторов, обеспечивающая достижение взаимопонимания и консенсуса при принятии решений относительно устраивающих всех способов урегулирования проблемных ситуаций [39].

**Неоднородный актор** – субъект, вовлеченный в урегулирование проблемной ситуации. Вовлеченность означает заинтересованность субъекта в изменении ситуации и обладание полезными, с точки зрения урегулирования ситуации, ресурсами [35].

**Ценностные ориентиры субъектов** - система принципов, взглядов и предпочтений, определяющая отношение к окружающей действительности, в том числе к проблемной ситуации. Ценностные ориентиры определяют разное отношение неоднородных акторов к одному и тому же событию или явлению.

**Консолидированная онтологическая модель** - единое понятийное пространство, необходимое для принятия неоднородными акторами ими всеми признаваемого решения в рамках урегулирования проблемной ситуации. Основой формирования консолидированной онтологической модели являются признаваемые неоднородными акторами определения сущностей. Любое исследование начинается с того, что даются определения важным, в рамках данного исследования, понятиям.

**Единое информационное пространство** - совокупность информационных систем, построенных на разнородных программно-аппаратных платформах, взаимодействующих на основе сетевых технологий между собой, образуя информационную инфраструктуру и Единое информационное пространство сложной системы.

**СПИСОК ЛИТЕРАТУРЫ**

1. Аврамчук, Е. Ф. Технология системного моделирования / Е. Ф. Аврамчук, А. А. Вавилов. - Москва: Машиностроение, 1988. - 520 с.
2. Аджиев, В. Миры о безопасности ПО: уроки знаменитых катастроф / В. Аджиев // Открытые системы. СУБД. - 1998, №6.
3. Андерсен, Б. Анализ основной причины. Упрощенные инструменты и методы / Б. Андерсен, Т. Фагерхоуд. - Милуоки, Висконсин: ASQ Quality Press, 1999. - 155 с.
4. Афанасьев, Ф. Управление проектами в стиле ДРАЙВ / Ф. Афанасьев. - Москва: Издательские решения, 2017. - 102 с.
5. Бежаева, О. Я. Информационная поддержка проектирования спецификаций требований пользователя / В. Е. Гвоздев, О. Я. Бежаева // Вестник Уфимского государственного авиационного технического университета. - 2014. - Т. 18, №5(66). - С. 57-61.
6. Бежаева, О. Я. Принципы и подходы к проектированию требований к программным продуктам и программным проектам // О. Я. Бежаева, В. Е. Гвоздев // Проблемы управления и моделирования в сложных системах: сб. науч. тр. (Самара, 2014 г.). - Самара: ИПУСС РАН, СНЦ РАН, 2014. - С. 196-203.
7. Бежаева, О. Я. Выявление противоречий в требованиях к программному продукту на основе исследования непрямых связей между ними / В. Е. Гвоздев, О. Я. Бежаева, Р. Р. Курунова // Программная инженерия. - 2015. - № 7. - С. 11-20.
8. Бежаева, О. Я. Оценивание линейных взаимосвязей параметров технических объектов при отсутствии корреляционной таблицы эмпирических данных / В. Е. Гвоздев, О. Я. Бежаева, А. С. Субхангулова // Вестник Уфимского государственного авиационного технического университета. - 2015. - Т.19, №4(70). - С. 106-117.

9. Бежаева, О. Я. Сравнительный анализ проектных решений по комплексу потребительских свойств и технических характеристик объектов / В. Е. Гвоздев, Б. Г. Ильясов, О. Я. Бежаева, Р. Р. Курунова // Информационные технологии. - 2016. - Т. 22. - № 10. - С. 764-770.
10. Бежаева, О. Я. Информационная поддержка планирования ресурсов для устранения дефектов на стадии реализации компонентов аппаратно-программных комплексов / В. Е. Гвоздев В. Е., О. Я. Бежаева, Субхангулова А. С. // Проблемы управления и моделирования в сложных системах: сборник трудов XVIII международной конференции. – Самара: ИПУСС РАН, 2016. – С. 271-276.
11. Бежаева, О. Я. Построение модели многосвязного объекта на основе совместного использования данных и экспертных оценок / В. Е. Гвоздев, Р. А. Мунасыпов, О. Я. Бежаева, Д. Р. Ахметова // Онтология проектирования. - 2019. - Т.9, №3. - С. 361-368.
12. Бежаева, О. Я. Анализ надежности информационных систем на основе концепции профилей сервисов / В. Е. Гвоздев, О. Я. Бежаева, Р. А. Насырова // Проблемы управления и моделирования в сложных системах: сб. науч. тр. (Самара, 2019 г.). - С. 21-25.
13. Бежаева, О. Я. Элементы системной инженерии: критические факторы программных проектов: учебное пособие / В. Е. Гвоздев, О. Я. Бежаева, Д. В. Блинова, Н. И. Ровнейко, К. В. Кириллов. Уфа: РИК УГАТУ, 2019. – 80 с.
14. Бежаева, О. Я. Формирование параметров модели управления проектом на основе линеаризации функциональных зависимостей / В. Е. Гвоздев, О. Я. Бежаева, Д. Р. Ахметова, Г. Р. Сафина // Онтология проектирования. - 2020. - Т.10 №4 (38). - С.527-539.
15. Бежаева, О. Я. Информационная поддержка проактивного управления функциональной безопасностью компонентов киберфизических систем / В. Е. Гвоздев, М. Б. Гузаиров, О. Я. Бежаева, Р. Р. Курунова, Р. А. Насырова // Моделирование, оптимизация и информационные технологии. - 2020. - Т. 8. - № 2(29). - С. 36-37.

16. Бежаева, О. Я. Модели возникновения ошибок на предпроектной стадии разработки компонент информационно-вычислительных систем / В. Е. Гвоздев, О. Я. Бежаева, Р. А. Насырова // Онтология проектирования. - 2020. - Т. 10. - №1. - С. 73-86.
17. Бежаева, О. Я. Обеспечение функциональной безопасности аппаратно-программных комплексов в условиях неопределенности среды использования / В. Е. Гвоздев, М. Б. Гузаиров, О. Я. Бежаева, А. С. Давлиева, Р. Р. Галимов // Моделирование, оптимизация и информационные технологии. - 2020. - №3.
18. Бежаева, О. Я. Методические основы управления функциональной безопасностью информационно-вычислительных компонентов на основе концепции «барьерное мышление» / О. Я. Бежаева, В. Е. Гвоздев, Н. И. Юсупова // Информационные системы и технологии. Достижения и перспективы: сб. науч. тр. (Сумгait, 2020 г.). – 2020, С. 340-342.
19. Бежаева, О. Я. Анализ влияния качества управления проектом на состояние функциональной безопасности аппаратно-программных комплексов на основе системного архетипа «предел роста» / О. Я. Бежаева, В. Е. Гвоздев, М. Б. Гузаиров // Моделирование, оптимизация и информационные технологии. - 2021. - № 9(3).
20. Бежаева, О. Я. Системные архетипы как методическая основа обеспечения функциональной безопасности аппаратно-программных комплексов / О. Я. Бежаева // International Journal of Open Information Technologies. - 2021. - Т. 9. - № 4. - С. 34-40.
21. Бежаева, О. Я. Оценка бюджета проекта по критериям удовлетворённости акторов / В. Е. Гвоздев, О. Я. Бежаева, Д. Р. Ахметова, Г. Р. Сафинова // Онтология проектирования. - 2021. - Т. 11. - № 3. - С. 382-392.
22. Бежаева, О. Я. Принципы и подходы к обеспечению функциональной безопасности компонентов вычислительно-коммуникационных систем / О. Я. Бежаева // International Journal of Open Information Technologies. - 2022. - Т. 10. - № 3. - С. 23-29.

23. Бежаева, О. Я. Обеспечение функциональной безопасности аппаратно-программных комплексов в условиях неопределенности среды использования / В. Е. Гвоздев, В. И. Васильев, М. Б. Гузаиров, О. Я. Бежаева // Моделирование, оптимизация и информационные технологии. - 2022. - №10 (2).
24. Бежаева, О. Я. Многоаспектное моделирование ситуаций в задачах обеспечения функциональной безопасности аппаратно-программных комплексов / В. Е. Гвоздев, О. Я. Бежаева, Д. Р. Сафинова // Онтология проектирования. - 2023. - Т.13, №1 (47) – С. 125 - 138.
25. Бежаева, О. Я. Анализ коренных причин возникновения дефектов при управлении сложными системами: учебное пособие / В. Е. Гвоздев, О. Я. Бежаева, А. С. Ракипова – Уфа: РИК УУНИТ, 2024. – 88 с.
26. Бежаева, О. Я. Мониторинг согласованности видений неоднородными акторами факторов проблемных ситуаций на основе статистической обработки экспертной информации / В. Е. Гвоздев, О. Я.Бежаева, Р. Р. Галимов, К.Ю. Тимофеева // Информационные технологии интеллектуальной поддержки принятия решений (Памяти проф. Н.И. Юсуповой) ITIDS'2024: Труды X Международной научной конференции. Уфа, 2024г. – Том 1. – С. 279-283.
27. Бежаева, О. Я. Оценка согласованности мнений неоднородных акторов на предпроектной стадии / В. Е. Гвоздев, О. Я. Бежаева, Р. Р. Галимов, К. Ю. Тимофеева // Онтология проектирования. - 2025. - Т. 15. - № 1 (55). - С.130-141.
28. Бежаева, О. Я. Количественные оценки степени согласованности точек зрения акторов / О. Я. Бежаева, А. С. Ракипова, К. Ю. Тимофеева // Молодежный Вестник УГАТУ. - 2024. - № 2 (31). - С. 16 - 19.
29. Бежаева, О. Я. Анализ структурной надежности многослойных взаимодействующих сетей передачи данных / О. Я. Бежаева, В. Е. Гвоздев, В. Е. Приходько [и др.] // Опыт и перспективы совершенствования систем связи и акустических испытаний современной техники : Сборник научных трудов научно-практической конференции, приуроченной к 130-летию со дня изобретения радио, Санкт-Петербург, 09 апреля 2025 года. – Санкт-Петербург: Военная академия связи им. Маршала Советского Союза С.М. Буденного, 2025. – С. 41-49.

30. Беркун, С. Искусство управления ИТ-проектами / С. Беркун. - Санкт-Петербург: Питер, 2014. - 698 с.
31. Бусленко, Н. П. Моделирование сложных систем / Н. П. Бусленко. - Москва: Наука, 1978 – 356 с.
32. Бэгьюли, Ф. Управление проектом / Ф. Бэгьюли. - Москва: Гранд-Фаир, 2002. - 208 с.
33. Быковский, С. В. Сопряженное проектирование встраиваемых систем (Hardware/Software Co-Design): учебное пособие / С. В. Быковский Я. Г. Горбачев, А. О. Ключев, А. В. Пенской, А. Е. Платунов. - Санкт-Петербург: Университет ИТМО, 2016. - 108 с.
34. Вентцель, Е. С. Теория вероятностей: учебник / Е. С. Вентцель. - 12-е изд. - Москва: Юстиция, 2018. - 658 с.
35. Виттих, В. А. Введение в теорию интерсубъективного управления / В. А. Виттих. - Самара: Самарский научный центр РАН, 2013. - 64 с.
36. Виттих, В. А. Неоднородный актор и повседневность как ключевые понятия эвергетики / В. А. Виттих. - Самара: Самарский научный центр РАН, 2014. - 12 с.
37. Виттих, В. А. Онтологии в интерсубъективных теориях / В. А. Виттих, А., М. В. Игнатьев, С. В. Смирнов // Мехатроника, автоматизация, управление. – 2012. - №5. - С. 69–70.
38. Виттих, В. А. Понятие интерсубъективности в эвергетике / В. А. Виттих // Онтология проектирования. - 2014. - №4 (14). - С. 90-97.
39. Виттих, В. А. Избранные труды по эвергетике (по материалам статей и докладов) / сост. С. Ю. Боровик, Т. В. Моисеева, С. В. Смирнов. - Самара: Издательство «Новая техника», 2022. - 420 с.
40. Гвоздев, В.Е. Анализ подходов к управлению функциональной безопасностью аппаратно-программных комплексов / В. Е. Гвоздев, Н. И. Юсупова, Л. Р. Черняховская, Д. Ф. Мамедов // Информационные системы и технологии. Достижения и перспективы: сб. науч. тр. (Сумгаит, 2019 г.). – 2019, С. 340-342.

41. Гельруд, Я. Д. Управление проектами: методы, модели, системы / Я. Д. Гельруд, О. В. Логиновский. - Челябинск: Издательский центр ЮУрГУ, 2015.
42. ГОСТ 51901.13-2005. Менеджмент риска. Анализ дерева неисправностей. - // Техэксперт: [сайт]. - URL: <http://docs.cntd.ru/document/gost-r-51901-11-2005> (дата обращения 29.09.2024).
43. ГОСТ 62502-2014. Менеджмент риска. Анализ дерева событий. - // Техэксперт: [сайт]. - URL: <http://docs.cntd.ru/document/1200114221> (дата обращения 29.09.2024).
44. ГОСТ Р 57100-2016. Системная и программная инженерия. Описание архитектуры. - Москва: Стандартинформ, 2019. - 36 с.
45. ГОСТ Р 56920-2016/ ISO/IEC/IEEE 29119 1:2013. Системная и программная инженерия. Тестирование программного обеспечения Часть 1. Понятия и определения. - Москва: Стандартинформ, 2016. - 53 с.
46. ГОСТ Р ИСО/МЭК 12207-2010. Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных систем. - Москва: Стандартинформ, 2012. - 105 с.
47. ГОСТ Р МЭК 61508-2 – 2012. Функциональная безопасность систем электрических, электронных, программируемых электронных, связанных с безопасностью. Требования к программному обеспечению. - Москва: Стандартинформ, 2014. -137 с.
48. ГОСТ 34.601-90. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания. - Москва: Стандартинформ, 2009. – 6 с.
49. ГОСТ 24.104-85. Единая система стандартов автоматизированных систем управления. Автоматизированные системы управления. Общие требования. - Москва: Стандартинформ, 2009. – 11 с.
50. ГОСТ Р ИСО / МЭК 25010-2015 «Информационные технологии. Системная и программная инженерия. Требования и оценка качества систем и программного обеспечения (SQuaRE). Модели качества систем и программных продуктов. - Москва: Стандартинформ, 2015. – 36 с.

51. ГОСТ Р 27.002-2015 Надежность в технике. Термины и определения. (Dependability in technics. Terms and definitions) - Москва: Стандартинформ, 2016. – 22 с.
52. ГОСТ Р 27.015-2019 / МЭК 60300-3-15:2009 Надежность в технике. Управление надежностью. Руководство по проектированию надежности систем (Dependability in technics. Dependability management. Guide for engineering of system dependability) - Москва: Стандартинформ, 2019. – 49 с.
53. ГОСТ 33707-2016 (ISO/IEC 2382:2015). Информационные технологии. Словарь. – М.: Стандартинформ, 2017. – 196 с.
54. Гузаиров, М.Б. Статистическое исследование территориальных систем: монография / Гузаиров М. Б. и др.; науч. ред. Гвоздев В. Е. – Москва: Машиностроение, 2008. – 186 с.
55. Гузаиров, М.Б. Элементы системной инженерии: методологические основы разработки программных систем на основе V-модели жизненного цикла: монография / М. Б. Гузаиров, В. Е. Гвоздев, Ильясов Б. Г., О. Я. Бежаева. – Москва: Машиностроение, 2013. – 180 с.
56. Дамбраускас, А. П. Симплексный поиск / А. П. Дамбраускас. - Москва: Энергия, 1979. - 176 с.
57. Демиденко, Е. З. Линейная и нелинейная регрессии / Е. З. Демиденко. - Москва: Финансы и статистика, 1981. - 304 с.
58. Дугин, А. Г. Сетевые войны. Аналитический доклад Александра Дугина при участии Валерия Коровина и Александра Бовдунова Изборскому клубу [Электронный ресурс]: URL: <https://izborsk-club.ru/2319> (дата обращения: 20.03.2025).
59. Жуков, И. С. Барьеры безопасности: понятие, классификация, концепции / И. С. Жуков // Безопасность труда в промышленности. – 2017. – № 5. – С. 49-56.
60. Зак, Ю. А. Принятие решений в условиях нечетких и размытых данных: Fuzzy-технологии / Ю. А. Зак. - Москва: Книжный дом "ЛИБРОКОМ", 2013. - 352 с.

61. Каган, А. М. Характеризационные задачи математический статистики / А. М. Каган, Ю. В. Линник, С. Р. Рао. - Москва: Наука, 1972. - 656 с.
62. Кобзарь, А.И. Прикладная математическая статистика. Для инженеров и научных работников. / А.М. Кобзарь. - Москва: ФИЗМАТЛИТ, 2006, С. 624-626.
63. Кириллов, Н. П. Концептуальная модель объекта ситуационного управления функциональным состоянием технических систем / Н. П. Кириллов // Искусственный интеллект и принятие решений. - 2012. - Т.4. - С. 61-75.
64. Кремер, Н. Ш. Эконометрика: учебник и практикум для вузов/ Н.Ш. Кремер. - Москва: Издательство Юрайт, 2024. - 308с.
65. Козлов, С. В. Проблемы интероперабельности в сетевентрических системах управления / С. В. Козлов, С. И. Макаренко, А. Я. Олейников, Д. В. Растворяев, Т. Е. Черницкая // Журнал радиоэлектроники. - 2019. - №12. - С.1-34.
66. Котов, С. Л. Информационно-аналитическая система оценивания трудозатрат и стоимости создания программных средств / С. Л. Котов, А. А. Демирский // Программные продукты и системы. - 2017. - №3(30). - С.469-473.
67. Криони, Н.К. Элементы системной инженерии. Технологии формирования требований к аппаратно-программным комплексам на основе экспертно-статистических методов: монография / В. Е. Гвоздев, Ильясов Б. Г., О. Я. Бежаева, Д. В. Блинова. – Москва: Машиностроение, 2017. – 295 с.
68. Кудж, С. А. Сетевентрическое управление и киберфизические системы / С. А. Кудж, В. Я. Цветков // Образовательные ресурсы и технологии. - 2017. - №2(19). - С. 86-92.
69. Кудж, С. А. Концепция сетевентрического управления сложной организационно-технической системой: монография. / С. А. Кудж, В. Я. Цветков, А. Н. Тихонов, А. Д. Иванников. – Москва: Макс Пресс, 2010. – 136 с.
70. Кузнецов, О. П. Человеческий фактор в управлении / О. П. Кузнецов, А. А. Кулинич, А. В. Марковский. - Москва: КомКнига, 2006. - 344 с.
71. Куликов, С. С. Тестирование программного обеспечения. Базовый курс. / С. С. Куликов. - Минск: Четыре четверти, 2017. - 312 с.

72. Линник, Ю. В. Метод наименьших квадратов и основы математико-статистической теории обработки наблюдений / Ю. В. Линник. - 2-е изд., доп. и испр. - Москва: Гос. Изд-во физ.-мат. лит., 1962. - 349 с.
73. Липаев, В. В. Надежность и функциональная безопасность комплексов программ реального времени / В. В. Липаев. - Москва: Институт программирования РАН, 2013. - 207 с.
74. Липаев, В. В. Надежность программных средств / В. В. Липаев. - Москва: Синтег, 1998. - 232 с.
75. Липаев, В. В. Технико-экономическое обоснование проектов сложных программных средств / В. В. Липаев. - Москва: Синтег, 2004. - 284 с.
76. Липаев, В. В. Функциональная безопасность программных средств / В. В. Липаев. - Москва: Синтег, 2004. - 348 с.
77. Липаев, В. В. Качество крупномасштабных программных средств / В. В. Липаев. - Москва-Берлин: Директ-медиа, 2015. - 231 с.
78. Майерс, Г. Дж. Надежность программного обеспечения / Г. Дж. Майерс. - Москва: Издательство Мир, 1980. - 359 с.
79. Макаренко, С. И. Модели интероперабельности информационных систем / С. И. Макаренко, А. Я. Олейников, Т. Е. Черницкая // Системы управления, связи и безопасности. - 2019. - №4. - С. 215-245.
80. Макаренко, С.И. Сетецентрическая война – принципы, технологии, примеры и перспективы: монография / С.И. Макаренко, М.С.Иванов. – Санкт-петербург, 2018. - 899 с.
81. Макконнелл, С. Сколько стоит программный проект / С. Макконнелл. - Санкт-Петербург: Питер, 2007. - 297 с.
82. Марков, А. С. Модели оценки и планирования испытаний программных средств по требованиям безопасности информации / А. С. Марков / Вестник МГТУ им. Н. Э. Баумана. - 2011. - С. 30-103.
83. Медведкова, И. В. Сравнительный анализ методов оценки стоимости проектов по разработке программного обеспечения / И. В. Медведкова, А. А. Иванов // International Journal of Humanities and Natural Sciences. - 2009. - №6(1). - С. 138-141.

84. Медоуз, Д. Азбука системного мышления / Д. Медоуз. - Москва: Бином. Лаборатория знаний, 2010. - 343 с.
85. Милошевич, Д. Набор инструментов для управления проектами / Д. Милошевич. - Москва: ДМК Пресс, 2008. - 729 с.
86. Мороз, Г. Концепция профилей в инженерии надежности программных систем / Г. Мороз, Г. Коваль, Т. Коротун // Математические машины и системы. - 2004. - №1. - С. 166-182.
87. Мостовой, А.Я. Имитационная математическая модель внешней среды в жизненном цикле бортового программного обеспечения управления космической платформой / А. Я. Мостовой // Компьютерная оптика, 2012. - Т.36, № 3. - С.412-418.
88. Мостовой А.Я. Управление сложными техническими системами: конструирование программного обеспечения спутников ДЗЗ: монография / А.Я. Мостовой. - Москва: ТЕХНОСФЕРА, 2016. – 352 с.
89. Мудров, В. И. Метод наименьших модулей / В. И. Мудров, В. Л. Кушков. - Москва: Радио и связь, 1983. - 304 с.
90. Муравьева-Витковская, Л.А. Моделирование интеллектуальных систем / Л. А Муравьева-Витковская – СПб.: НИУ ИТМО, 2012. – 145с.
91. Нагибин, С. Я. Технологическая безопасность программирования - новая проблема в области создания информационных систем / С. Я. Нагибин, Б. П. Пальчун, Л. М. Ухлинов // Информационное общество. - 1995. - Т. 6. - С. 45-49.
92. О'Коннор, Дж. Искусство системного мышления: Необходимые знания о системах и творческом подходе к решению проблем / Дж. О'Коннор. - Москва: Альпина Паблишер, 2013. - 254 с.
93. Отчет об устойчивом развитии Группы «ЛУКОЙЛ». 2023 – 115с.  
URL:<https://lukoil.ru/FileSystem/9/666712.pdf> (дата обращения 01.12.2024)
94. Платунов, А. Е. Высокоуровневое проектирование встраиваемых систем (часть 1): учебное пособие / А. Е. Платунов, Н. П. Постников. – СПб.: НИУ ИТМО, 2011. – 121 с.

95. Платунов, А. Е. Высокоуровневое проектирование встраиваемых систем (часть 2): учебное пособие / А. Е. Платунов, Н. П. Постников. – СПб.: НИУ ИТМО, 2013. – 172 с.
96. Пугачев, В. С. Теория вероятностей и математическая статистика / В. С. Пугачев. - 2-е изд., испрavl. и доп. - Москва: Физматлит, 2002. - 496 с.
97. Райков, А. Н. Конвергентное управление и поддержка решений / А. Н. Райков. - Москва: Издательство Икар, 2009. - 245 с.
98. Райков, А. Н. Сетевая экспертная поддержка решений / А. Н. Райков // Управление большими системами. - 2010. - Т. 30. № 1. - С. 758-772.
99. Рот, А. Внедрение и развитие Индустрии 4.0. Основы, моделирование и примеры из практики / А. Рот. - Москва: Техносфера, 2017. - 294 с.
100. Руководство к своду знаний по управлению проектами (PMBOK Guide) / Project Management Institute, 2012. - 585 с.
101. Рябинин, И. А. Надежность и безопасность структурно-сложных систем / И. А. Рябинин. - 2-е изд., перераб. и доп. - Санкт-Петербург: Изд-во СПбГУ, 2007. - 276 с.
102. Свидетельство об офиц. рег. программы для ЭВМ № 2020667386 Программа для формирования сбалансированной системы характеристик проекта / О. Я. Бежаева, В. Е. Гвоздев, Д. Р. Ахметова, Г. Р. Сафинова // Роспатент.: Зарег. В Реестре программ для ЭВМ 23.12.2020 г.
103. Свидетельство об офиц. рег. программы для ЭВМ № 2021616053 Программа для оценки критического пути по показателю неопределенности / О. Я. Бежаева, В. Е. Гвоздев, Д. Р. Ахметова, К. В. Седлецкий // Роспатент.: Зарег. В Реестре программ для ЭВМ 15.04.2021 г.
104. Свидетельство об офиц. рег. программы для ЭВМ № №2023667695 Идентификация проблемных ситуаций при управлении проектами / О. Я. Бежаева, Хуснутдинов Х.В. // Роспатент.: Зарег. В Реестре программ для ЭВМ 10.08.2023 г.
105. Свидетельство об офиц. рег. программы для ЭВМ № 2025668826 Программа для анализа моделей проектируемых программных компонентов по критериям функциональной надежности / О. Я. Бежаева, В. Е. Гвоздев, А. И. Газизов // Роспатент.: Зарег. В Реестре программ для ЭВМ 18.07.2025 г.

106. Свидетельство об офиц. рег. программы для ЭВМ №2025680058 Программа для расчета степени согласованности неоднородных акторов на основе ранговых оценок проблемной ситуации / Бежаева О. Я., Гвоздев В. Е., Тимофеева К.Ю. // Роспатент. М.: Зарег. в реестре программ для ЭВМ 01.08.2025 г.
107. Сенге, П. М. Пятая дисциплина / П. М. Сенге. - Москва: Олимп-Бизнес, 2003. - 408 с.
108. Симонова, Е.В. Моделирование информационных систем. Часть I. Адаптивное управление сложными системами на основе мультиагентных технологий: учебное пособие / Е.В. Симонова. – Самара: Издательство Самарского университета, 2022. – 204 с.
109. Смирнов, Н. В. Курс теории вероятностей и математической статистики для технических приложений / Н. В. Смирнов, И В. Дунин-Барковский. - Москва: Наука, 1969. - 512 с.
110. Соболь, И. М. Численные методы Монте-Карло / И. М. Соболь. - Москва: Наука, 1973. - 312 с.
111. Скobelев, П.О. Как управлять сложными системами? Мультиагентные технологии для создания интеллектуальных систем управления предприятиями / Г. А. Ржевский, П. О. Скobelев. – перевод с англ. – WIT Press, 2015. – 290 с.
112. Трусов, П. В., Введение в математическое моделирование: учебное пособие / П. В. Трусов, В. Н. Ашихмин, М. Б. Гитман [и др.]. - Москва: Логос, 2005. - 440 с.
113. Тимофеев, А. Н. Почему падают ИТ-проекты? / А. Н. Тимофеев // Практика проектирования систем. - 2017. - С. 2-12.
114. Трутнев, Д. Р. Архитектуры информационных систем. Основы проектирования / Д. Р. Трутнев. - Санкт-Петербург: НИУ ИТМО, 2012. - 125 с.
115. Управление проектами: основы профессиональных знаний. Национальные требования к компетентности специалистов по управлению проектами. Версия 3.0 / под ред. В. И. Воропаева. - Москва: Проектная практика, 2010. - 256 с.
116. Черкесов, Г.Н. Надежность аппаратно-программных комплексов. / Г.Н. Черкесов. - СПб.: Питер, 2005. - 147 с.

117. Черняк, Л. Киберфизические системы. Cyber-Physical System (CPS). К чему приведет слияние интернета, людей, вещей и сервисов. / Л. Черняк. // [сайт]. - URL: <http://www.tadviser.ru/a/374827> (дата обращения 12.02.2020).
118. Черняховская, Л.Р., Методы и модели поддержки принятия решений при управлении инновационными проектами в производственно-экономических системах: монография / Л.Р. Черняховская, В. И. Васильев, Вульфин А.М., В. Е. Гвоздев, О. Я. Бежаева, Н. О. Никулина, А. И Малахова. – Москва: Издательский дом Академии Естествознания, 2021. – 230 с.
119. Черняховская, Л.Р., Информационная поддержка формирования параметров программных проектов / О. Я. Бежаева, В. Е. Гвоздев, Л. Р. Черняховская, Н. О. Никулина // Информационные технологии в управлении: материалы всероссийской конференции с международным участием (Санкт-Петербург, 2020 г.). – 2020. - С. 112-115.
120. Черняховская, Л.Р. Эвергетика как методологическая основа управления выявлением дефектов на предпроектной стадии жизненного цикла систем обработки данных / В. Е. Гвоздев, Л. Р. Черняховская, Д. В. Блинова // Онтология проектирования. – 2018. – Т. 8. №1(27). – С. 22–29.
121. Шведин, Б. Я. Онтология предприятия: экспириентологический подход. Технология построения онтологической модели предприятия на основе анализа и структурирования живого опыта / Б. Я. Шведин. - Москва: ЛЕНАНД, 2010. - 240 с.
122. Шубинский И.Б. Функциональная надежность информационных систем. Методы анализа / И.Б. Шубинский. – М.: «Журнал Надежность», 2012, – 296 с.
123. Шубинский И.Б. Структурная надежность информационных систем. Методы анализа / И.Б. Шубинский. – М.: «Журнал Надежность», 2012, – 216 с.
124. 4CIO: учебник / под ред. С. Кирюшина, К. Зимина. - 2 изд. - Москва: 4CIO, 2013 - 700 с.
125. 4CIO: учебник / под ред. С. Кирюшина. - Спецвыпуск к ПВ-2018. - Москва: 4CIO, 2018. - 230 с.

126. 1471-2000. IEEE Recommended Practice for Architectural Description of Software-Intensive Systems // IEEE Standards Association: [сайт]. - URL: <https://standards.ieee.org/standard/1471-2000.html> (дата обращения 08.10.2024)
127. A guide to the project management body of knowledge (PMBOK guide): 5th edition. - Project Management Institute, 2012. - 585 p.
128. Ahuett-Garzaa, H. A brief discussion on the trends of habilitating technologies for Industry 4.0 and Smart manufacturing / H. Ahuett-Garzaa, F. Kurfessb // Manufacturing Letters. - 2018. - №15. - P. 60-63.
129. Andersen, B. Root cause analysis. Quality Press / B. Andersen, T. Fagerhaug // . – 2006.
130. Anu, V. K. Using Human Error Models to Impove the Quality of Software Requirements / V. K. Anu; North Dakota State University of Agriculture and Applied Science. - Fargo, 2018. - 119 p.
131. Anu, V. K. Detection of Requirement Errors and Faults via a Human Error Taxonomy: A Feasibility Study / V. Anu, W. Hu, J. Carver, [et al.]. - 2016. DOI:10.1145/2961111.2962596
132. Bezhaeva, O.Y. The Method for Linear Regression Models Constructing Based on the Sharing of Measured Data and Expert Assessments / V. E. Gvozdev, O. Y. Bezhaeva, D. R. Achmetova, A. A. Levkov // Atlantis Highlights in Computer Sciences. – 2019. - Vol. 3. – P. 89-93.
133. Bezhaeva, O.Y. The Model of the Innovative Project as the Multivariable Control Object / V. E. Gvozdev, O. Y. Bezhaeva, D. V. Blinova, D. R. Akhmetova // Advances in Intelligent Systems Research. – 2020. - Vol. 174. - P. 278-282.
134. Bezhaeva, O. Y. Formation of Balanced System of Program Project Characteristics / O. Y. Bezhaeva, V. E. Gvozdev, D. R. Akhmetova, G. Safyna // Advances in Intelligent Systems Research. – 2020. - Vol. 174. - P. 302-306.
135. Bezhaeva, O. Y. The studies of strategies for ensuring the functional safety of hardware and software complexes based on system archetypes and architecture models / O. Y. Bezhaeva, V. E. Gvozdev // Proceedings of the International Conference on Electrotechnical Complexes and Systems, ICOECS (Ufa, Russia, November 2021). Publisher: IEEE. – 2021. - P. 171-175.

136. Bezhaeva, O.Y. Construction of Signal Transmission Structures in Information Systems Based on Conjunction Schemes / O. Y. Bezhaeva, V. E. Gvozdev, Galimov R.R. // Proceedings of the 2023 International Ural Conference on Electrical Power Engineering, UralCon, (Magnitogorsk, Russia, September 2023). Publisher: IEEE. – 2023. – P. 1-5.
137. BowTieXP. The next generation BowTie methodoloy tool / BowTie Methodology Manual Revision 15 (27 March 2015). - 64 p.
138. Braun W. The System Archetypes / W. Braun. - 2002. - 25 p.
139. Brooks, F. P. No Silver Bullet: Essence and Accidents of Software Engineering / Brooks F. P. // Computer. - 1987. - №20 - p. 10-19.
140. Cohen L. Quality Function Deployment and Six Sigma: A QFD Handbook / L. Cohen, J. P. Ficalora - 2nd Edition, Pearson Education, Inc., 2010. - 527 p.
141. Carver, J. C. Defect prevention in requirements using human error information: An empirical study. / J. C. Carver, W. Hu, V. Anu, G. Walia, G. Bradshaw // Requirements Engineering: Foundation for Software Quality - 23rd International Working Conference, REFSQ, – 2017. - p. 61-76
142. Case, D. Fuzzy Cognitive Map to Model Project Management Problems / D. Case, C. Stylios // Proceedings of 35th Annual Conference of the North American Fuzzy Information Processing Society NAFIPS, - 2016. - P. 61-76
143. CHAOS MANIFESTO. The Standish Group International, Inc., 2013. <https://larlet.fr/static/david/stream/ChaosManifesto2013.pdf>
144. CHAOS Report. The Standish Group International, Inc., 2015. [https://www.standishgroup.com/sample\\_research\\_files/CHAOSReport2015-Final.pdf](https://www.standishgroup.com/sample_research_files/CHAOSReport2015-Final.pdf)
145. CHAOS Report. The Standish Group International Inc., 2018. [https://www.standishgroup.com/sample\\_research\\_files/CHAOSReport2018-Final.pdf](https://www.standishgroup.com/sample_research_files/CHAOSReport2018-Final.pdf)
146. Christian M. T. Using TRIZ to invent failures - concept and application to go beyond traditional FMEA / M. T. Christian, F. Zeihsel, S. Visnepolschi [et al.] // Procedia Engineering 131. -2015.- p. 426 – 450
147. Cognitive Approach to Root Cause Analysis for Improvement Quality of Life: A Case Study for IT industry // International Journal of Informative and Futuristic Research (Online). - 2013. - №1

148. Cortellessa, V. A modeling approach to analyze the impact of error propagation on reliability of component-based systems / V. Cortellessa, V. Grassi // Lecture Notes in Computer Science. - 2007. - P. 140-156.
149. DeMarco, T. The human factor: successful projects and teams / T. DeMarco. - Symbol-Plus, 2014. - 288 p.
150. Duphily, R. Root Cause Investigation Best Practices Guide / R. Duphily // AEROSPACE REPORT NO. TOR-2014-02202, May 30, 2014 – 95 p.
151. Embray, D. Understanding Human Behaviour and Error / D. Embray. - Human Reliability Associates, 2005. - 10 p.
152. ESA PSS-05-10. Guide to software verification and validation. - ESA Board for Software Standardization and Control (BSSC), 1995. - 117 p.
153. ESA PSS-05-11. Guide to software quality assurance. - 1995. - 55 p.
154. ESA-PSS-05-02. Guide to the user requirements definition phase. - 1995. - 55 p.
155. Gvozdev, V. E. Reliability analysis of information systems based on the concept of service profiles / V.E. Gvozdev, Bezhaeva O.Y., L.R. Chernyakhovskaya, Nasyrova R.A. // Proceedings of the 2019 XXI International Conference Complex Systems: Control and Modeling Problems, CSCMP (Samara, Russia, 2019). –Publisher: IEEE, - 2019. - P. 1-6.
156. Gvozdev, V. E. Ensuring the Functional Safety of the Distributed Dynamic Systems Components in the Conditions of Uncertainty of the Environment Use / O. Y. Bezhaeva, V. E. Gvozdev, M. B. Guzairov, A. S. Davlieva, R.R. Galimov // Proceedings of the International Conference on Electrotechnical Complexes and Systems, ICOECS (Ufa, Russia, 2020). - Publisher: IEEE, -2020. - P. 232-237.
157. Geisberger, E. Living in a networked world. Integrated research agenda Cyber-Physical Systems / E. Geisberger, M. Broy. - Acatech STUDY, 2015. - 291 p.
158. Getting to Grips with Human Factors in Drilling Operations / J. L. Thorogood, K. Lauche, M. Crichton [et al.]. - Society of Petroleum Engineers, 2015.
159. Guidance on using Tripod Beta in the investigation and analysis of incidents, accidents and business losses / TRIPOD BETA. - London: Energy Institute, 2015. - 19 p.

160. Gursimran, S. W. A systematic literature review to identify and classify software requirements errors / S. W. Gursimran, J. C. Carver. - Information and Software Technology, 2007. - 1087-1109 p.
161. Herzwurm, G. Usage of Quality Function Deployment in Europe: State of the art and selected case studies / G. Herzwurm, S. Schockert, P. Zimmerman. - University of Stuttgart, 2007. - 21 p.
162. Huang, F. Software defect prevention based on human error theories / F. Huang, B. Liu // Chinese Journal of Aeronautics. - 2017. - №30. - p. 1054-1070.
163. ICB IPMA Competence Baseline Version 3.0 International Project Management Association / G. Caupin, H. Knoepfel [et al.]. - IPMI, 2006. – 200 p.
164. Industrie 4.0 Maturity Index Managing the Digital Transformation of Companies / G. Schuh, R. Anderl, J. Gausemeier [et al.]. - Acatech STUDY, 2017.
165. Industrie 4.0 Maturity Index Managing the Digital Transformation of Companies / G. Schuh, R. Anderl, J. Gausemeier [et al.]. - 2nd edition. - Acatech STUDY, 2020. - 64 p.
166. ISO 21500:2012. Guidance on project management. - International Organization for certification. <https://www.iso.org/standard/50003.html>
167. ISO/IEC/IEEE 29148:2018, Systems and Software Engineering - Life cycle processes - Requirements Engineering. <https://www.iso.org/standard/72089.html>
168. ISO/IEC/IEEE 15288:2023, Systems and software engineering - System life cycle processes. <https://www.iso.org/standard/81702.html>
169. IEEE Std 26514-2010, IEEE Standard for Adoption of ISO/IEC 26514:2008 Systems and Software Engineering Requirements for Designers and Developers of User Documentation, – IEEE, 2011. - 72p.
170. IEEE Std 730-2014, IEEE Standard for Software Quality Assurance Processes. – IEEE, 2015. - 138p.
171. IEEE Std 1012-2016, IEEE Standard for System, Software, and Hardware Verification and Validation. – IEEE, 2017. – 260 p.
172. ISO/IEC/IEEE 42020:2019, Software, systems and enterprise — Architecture processes. <https://www.iso.org/standard/68982.html>

173. Jaiswal E.S. A Case Study on Quality Function Deployment (QFD) // Journal of Mechanical and Civil Engineering. 2012. - № 6 (3). - P. 27–35.
174. Kantorowitz, E. The performance of the N-fold requirement inspection method / E. Kantorowitz, A. Guttman, L. Arzi // Requirements Engineering. - 1997. - 2nd edition. - №3. - P. 152-164.
175. Kazman, R. Understanding Patterns for System-of-System integration / R. Kazman, C. Nielsen, K. Schmid. - Software Engineering Institute and Acquisition Practices, Technical Node, 2013. - 25 p.
176. Kaur, K. System Reliability, Availability, and Maintainability. / K. Kaur // SSRN Electronic Journal. -2023.
177. Khater, H. A. A Proposed Technique for Software Development Risks Identification by using FTA Model / H. A. Khater, A. B. Mohamed, S. M. Kamel // World Academy of Science, Engineering and Technology International Journal of Computer and Information Engineering. - 2013. - №7. - P. 105-111.
178. Klein, G. Anticipatory Thinking / G. Klein, D. Snowden, L. P. Chew // Proc. International NDM Conf (Pacific Grove, 2007). - Pacific Grove, 2007. - P. 1-7.
179. Kleppe, A. G. MDA Explained: The Model Driven Architecture: Practice and Promise / A. G. Kleppe, J. Warmer – Boston: Addison-Wesley Longman Publishing Co., 2003. - 170 p.
180. Lanubile, F. Experimenting with error abstraction in requirements documents / F. Lanubile, F. Shull, V. Basili // Proc. of the 5th Int. Symposium on Software Metrics (Bethesda, Maryland, 1998). - 1998 - P. 114-121.
181. Laprie, J. C. Software reliability trend analyses from theoretical to practical considerations / K. Kanoun, J. C. Laprie // IEEE Transactions on Software Engineering. - 1994. - Vol. 20, № 9. p. 740-747.
182. Laprie, J. C. A. Basic concepts and taxonomy of dependable and secure computing / A. Avizienis, J. C. Laprie, B. Randell, C. Landwehr // IEEE Transactions on Dependable and Secure Computing. - 2004. - Vol. 1, № 1. p. 11-33.
183. Lee, E. A. Cyber Physical Systems: Design Challenges / E. A. Lee // Electrical Engineering and Computer Sciences. - 2008. - P. 363-369.

184. Lee, E. A. Fault tolerance, principles and practice / E. A. Lee, T. Anderson // Springer Sci. Bus. Media. - 2012. - №3. – 320 p.
185. Living in a networked world. Integrated research agenda Cyber-Physical Systems (agendaCPS) / Eva Geisberger, Manfred Broy [et al.]. - Acatech STUDY, 2015. - 293 p.
186. Loo, J. Mobile Ad Hoc Networks. Current Status and Future Trends / J. Loo, J. Mauri, J. Ortiz // AUERBACH PUBLICATIONS, 2015. - 528 p.
187. Liu, X. Software Quality Function Deployment / X. Liu // IEEE Potentials, Institute of Electrical and Electronics Engineers. 2000. P. 14 -16.
188. Morozov, A. Dual Graph Error Propagation Model for Mechatronic System Analysis. / A. Morozov, K. Janschek // IFAC Proceedings. – 2011. Vol. 44, 9893-9898.
189. Morozov, A. Model-based Dependability Analysis of Fault-tolerant Inertial Navigation System: A Practical Experience Report / M. Steurer, A. Morozov, K. Janschek, K. P. Neitzke // IFAC-PapersOnLine. - 2019.
190. Nassar, D. Error Propagation Analysis of Software Architecture Specifications / D. Nassar [et al.] // Communication. - 2006. - №1. - P. 496-501.
191. New tools for failure and risk analysis anticipatory failure determination (AFD) and the theory of scenario structuring / S. Visneposchi, B. Zlotin, S. Kaplan, A. Zusman. - Ideation Intl Inc., 1999. - 86 p.
192. Nunes, D. A Practical Introduction to Human-in-the-Loop Cyber-Physical Systems / D. Nunes, J. Sa Silva, F. Boavida. - John Wiley & Sons Ltd., 2018.
193. Parnas, D. L. The Role of Inspection in Software Quality Assurance / D. L. Parnas, M. Lawford // IEEE Transactions on Software Engineering, 2003. - №29. - P. 674-676.
194. Pelaez, C. E. Using fuzzy cognitive maps as a system model for failure modes and effects analysis / C. E. Pelaez, J. B. Bowles // Information Sciences, 1996. - №88(1). - P. 177-199.
195. Pentti, H. Failure Mode and Effects Analysis of Software-Based Automation Systems / H. Pentti, H. Atte. - STUK-YTO-TR190, 2002. - 36 p.

196. Pramono, E. Defect Analysis Using Failure Mode and Effects Analysis and Fault Tree Analysis / E. Pramono, S. Noya, Y. Ekawati // Journal of Industrial Engineering: Application and Research (SAKTI), 2024. Vol. 4, №1 - P.17-29
197. Privalov A. Yu. Hybrid Model of Human Mobility for DTN Network Simulation / A. Yu. Privalov, A. A. Tsarev. // In Proceedings of 30th European Conference on Modelling and Simulation (ECMS2016), 2016. - P. 419-424
198. Perneger, T. V. The Swiss cheese model of safety incidents: Are there holes in the metaphor? / T. V. Perneger // BMC Health Services Research. - 2005. - № 5(1). [https://www.researchgate.net/publication/7488318\\_The\\_Swiss\\_cheese\\_model\\_of\\_safety\\_incidents\\_Are\\_there\\_holes\\_in\\_the\\_metaphor/](https://www.researchgate.net/publication/7488318_The_Swiss_cheese_model_of_safety_incidents_Are_there_holes_in_the_metaphor/)
199. Perini, A. Engineering requirements for adaptive systems. / M. Morandini, L. Penserini, A. Perini, A. Marchetto // Requirements Engineering. – 2017. - № 22(1). P.77-103.
200. Reason, J. Human Error / J. Reason. - Cambridge University Press, 1990. - 302 p.
201. Reason, J. Revisiting the "Swiss Cheese" Model of Accidents / J. Reason, E, Hollnagel, J. Paries. - European Organization for the Safety of Air Navigation, 2006. - 25 p.
202. Rung, M. Software Reliability Theory /Michael Rung, Tsong Lyu // Encyclopedia of Software Engineering. The Chinese University of Hong Kong, 2002. - 43 p.
203. Shappell S. A. The Human Factors Analysis and Classification System - HFACS. Final Report / S. A. Schapell, D. A. Weigmann. - U. S. Department of Transportation, Federal Aviation Administration, 2000.
204. Silva, R. Anticipatory Failure Determination (AFD) for product reliability analysis: A comparison between AFD and Failure Mode and Effects Analysys (FMEA) for identifying potential failure modes / R. Silva, M. Carvalho. Federal Technological University of Parana (UTFPR). - Curitiba, 2019. - 24 p.
205. Soltanmohammadi, S. Main human factors affecting information system security / S. Soltanmohammadi, S. Asadi, N. I. Norafida // Interdisciplinary Journal of Contemporary Research in Business. - 2013. - №5(7). - P. 329-354.

206. Soni, R. Cognitive approach to root cause analysis for improving quality of life: a case study for IT Industry / R. Soni, A. Preet // International Journal of informative and futuristic research, 2013. - №1 - P. 8.
207. Srivastava, A. Analysis of Software Reliability Data using Exponential Power Model / Ashwini Kumar Srivastava, Vijay Kumar // International Journal of Advanced Computer Science and Applications (IJACSA), 2011, Vol.2, № 2, pp. 38-45.
208. Sunday, E. Extension and Modification of Anticipatory Failure Determination Approach based on I-TRIZ. Master Thesis / E. Sunday. - 2014. - 110 p.
209. Systems Engineering Guide for Systems of Systems / Washington: ODUSD(A&T) SSE, 2008. - 135 p.
210. Travis, G. How the Boeing 737 Max Disaster Looks to a Software Developer / G. Travis // URL: <https://spectrum.ieee.org/aerospace/aviation/how-the-boeing-737-max-disaster-looks-to-a-software-developer>
211. Tolo, Silvia. Fault Tree Analysis Including Component Dependencies / Silvia Tolo, John Andrews// IEEE Transactions on Reliability №73 – 2024. P. 413-421.
212. Visnepolschi, S. Using TRIZ to invent failures - concept and application to go belong traditional FMEA / C. Thurnes, F. Zeihsel, S. Visnepolschi, F. Hallfell // Procedia Engineering. - 2015. - p. 426-450. - URL: <https://www.sciencedirect.com/>
213. Visnepolschi, S. New Tools for Failure and Risk Analysis / S. Kaplan, S. Visnepolschi, B. Zlotin, A. Zusman // Ideation International Inc., 2005. 68 p.
214. Verzola, I. A Predictive Approach to Failure Estimation and Identification for Space Systems Operations / I. Verzola, A. E. Lagny, J. Biswas // Proc. 13th international conference on space operations (Pasadena, California, 2014).
215. Volk, Matthias. SAFEST: Fault Tree Analysis Via Probabilistic Model Checking / M. Volk, S. Falak S, J. P. Katoen, M. Stoelinga // 2024 Annual Reliability and Maintainability Symposium (RAMS) – 2024. P. 1-7.
216. Zhu, Y. M. Failure-modes-based software reading. Chapter 2: Software Failure Mode and Effects Analysis. – 2017. № XI. p.7-15.

## Приложение А

(справочное)

### Копии актов внедрения результатов диссертационной работы



Общество с ограниченной ответственностью  
«Научно-производственная фирма «ЭИТЭК»  
117292, Москва, ул. Ивана Бабушкина, д. 10  
Телефон: +7 (495) 926-10-80  
E-mail: [mail@eitek.ru](mailto:mail@eitek.ru) | [www.eitek.ru](http://www.eitek.ru)  
ИИН 7713041098 / КПП 772801001  
р/с 40702810840020100273 в ПАО Сбербанк  
к/с 30101810400000000225, БИК 044525225



#### Акт

об использовании  
результатов диссертационной работы Бежаевой Оксаны Яковлевны  
«Методологические основы обеспечения функциональной надежности  
информационных систем на предпроектной стадии», представленной на соискание  
ученой степени доктора технических наук

Настоящим актом подтверждаем, что результаты, полученные Бежаевой Оксаной Яковлевной в диссертационной работе на тему «Методологические основы обеспечения функциональной надежности информационных систем на предпроектной стадии», были внедрены в ООО «Научно-производственная фирма ЭИТЭК», «НПФ ЭИТЭК» – компания-разработчик современных инжиниринговых решений для промышленных предприятий.

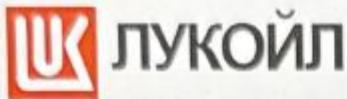
В ходе внедрения:

- применены разработанные принципы и подходы к обеспечению функциональной надежности, что позволило выявить факторы функциональной надежности локальных информационных систем при формировании единого информационного пространства компании НПФ ЭИТЭК;
- проведен кластерный анализ поступивших запросов в службу поддержки с целью выявления и систематизации ошибок, с которыми сталкиваются пользователи, на основе выделенных кластеров выполнен анализ коренных причин (RCA) и анализ деревьев отказов (FTA), обеспечивший комплексное изучение факторов;
- на основе полученных результатов разработаны типовые инструкции по сопровождению локальных информационных систем.

После внедрения результатов, полученных в диссертационной работе Бежаевой Оксаны Яковлевны, количество отказов в локальных информационных системах сократилось более чем в три раза, среднее время восстановления уменьшилось на 60 %, что свидетельствует о повышении функциональной надежности информационных систем, используемых в компании, и подтверждает высокую эффективность и практическую значимость предложенных подходов.

Директор Инжинирингового центра  
ООО «НПФ ЭИТЭК» в г. Уфе, к.т.н.

Сафаров Д.О.



ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ  
«ЛУКОЙЛ - Центрнефтепродукт»



А.Н. Нестеренко

2025г.

Акт  
об использовании  
результатов диссертационной работы Бежаевой Оксаны Яковлевны  
«Методологические основы обеспечения функциональной надежности  
информационных систем на предпроектной стадии», представленной на соискание  
ученой степени доктора технических наук

Настоящим актом подтверждаем, что результаты, полученные Бежаевой Оксаной Яковлевной в диссертационной работе на тему «Методологические основы обеспечения функциональной надежности информационных систем на предпроектной стадии» были апробированы в ООО «ЛУКОЙЛ-Центрнефтепродукт».

Применение разработанной в диссертационной работе Бежаевой О.Я. методики формальной оценки возможности реализации проекта позволяет получить экономический эффект за счет сокращения сроков предпроектной стадии (более 10%) и повышения обоснованности параметров проекта.

И.о. начальника отдела ИТО и связи

В.Н. Апанасенко



Акционерное общество  
**Научно-производственное предприятие**  
**"Полигон"**  
450015, Республика Башкортостан, г. Уфа,  
ул. Карла Маркса, 37  
Почтовый адрес: 450000, г. Уфа, а/я 1262  
Тел.: (347) 292-09-90  
<http://www.plgn.ru>; e-mail: [info@plgn.ru](mailto:info@plgn.ru)  
ОКПО 12688272 ОГРН 1030203893631  
ИНН/КПП 0274013026/027501001

**УТВЕРЖДАЮ**

Генеральный директор

АО НПП «Полигон»

И.А. Махов



2 сентября 2025 г.

№ \_\_\_\_\_

На № \_\_\_\_\_ от \_\_\_\_\_

**Акт****об использовании**

результатов диссертационной работы Бежаевой Оксаны Яковлевны  
«Методологические основы обеспечения функциональной надежности  
информационных систем на предпроектной стадии», представленной на  
соискание ученой степени доктора технических наук

Настоящий акт составлен о том, что результаты диссертационной работы  
Бежаевой Оксаны Яковлевны, а именно:

- метод структурного моделирования последствий отказов функциональных компонент информационных систем на основе схем сопряжения и таблиц истинности;
- методика анализа моделей проектируемых компонентов информационных систем по критериям функциональной надежности;
- программное обеспечение для анализа моделей проектируемых компонентов информационных систем по критериям функциональной надежности [Св-во об офиц. регистрации программы для ЭВМ №2025668826. Программа для анализа моделей проектируемых программных компонентов по критериям функциональной надежности / **Бежаева О. Я.**, Гвоздев В. Е., Газизов А. И. // Роспатент. М.: Зарег. в реестре программ для ЭВМ 18.07.2025г.] внедрены и используются в ОАО Научно-производственном предприятии «Полигон».

Результаты позволяют провести оценку возможных последствий отказов компонентов по различным критериям функциональной надежности на предпроектной стадии, обоснованно распределить ресурсы на обеспечение функциональной надежности, избавиться от трудоемкого и затратного по времени процесса ручного построения структурных моделей отказов.

Главный конструктор

Хомский В.Н.



ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ  
«ГАЗПРОМНЕФТЬ – ЦИФРОВЫЕ РЕШЕНИЯ»  
(ООО «ГАЗПРОМНЕФТЬ – ЦР»)

Адрес для корреспонденции:  
ул. Кинская, д. 5, кв. 4, Санкт-Петербург, 198084  
ОКПО 85683109, ОГРН 1087746449812.  
ИНН 7725024630, КПП 781901001  
Тел.: +7 812 448-24-01  
e-mail: ds-info@gazprom-neft.ru  
ds.gazprom-neft.ru

**Акт  
об использовании**

результатов диссертационной работы Бежаевой Оксаны Яковлевны  
«Методологические основы обеспечения функциональной надежности  
информационных систем на предпроектной стадии», представленной на  
соискание ученой степени доктора технических наук

Настоящий акт составлен о том, что результаты диссертационной работы  
Бежаевой Оксаны Яковлевны, а именно:

- оценка согласованности мнений неоднородных акторов на основе  
системных архетипов [Гвоздев В. Е., Бежаева О. Я., Галимов Р. Р., Тимофеева К.  
Ю. Оценка согласованности мнений неоднородных акторов на предпроектной  
стадии // Онтология проектирования. – 2025. – Т.15, №1 (55) – С. 130 - 141]

внедрены и используются как методическая основа для систематического  
мониторинга и анализа согласованности мнений внутри проектной команды в  
ходе проектирования.

Применение предложенной схемы позволяет сократить длительность  
предпроектной стадии разработки информационных систем за счёт ускоренной  
выработки консолидированного мнения о внешнем облике информационной  
системы, а также обеспечивает повышение эффективности коммуникаций между  
проектными командами и заинтересованными сторонами.

Руководитель направления  
Управление технологического развития  
ООО «ГПН – ЦР»



**ООО «ГАЗПРОМНЕФТЬ – ЦР»**



Утверждаю  
Проректор по развитию  
образования ФГБОУ ВО  
Уфимский университет  
науки и технологий  
Рахманова Ю.В.

«20» мая 2025г.

Акт

об использовании  
результатов диссертационной работы Бежаевой Оксаны Яковлевны  
«Методологические основы обеспечения функциональной надежности  
информационных систем на предпроектной стадии», представленной на соискание  
ученой степени доктора технических наук

Мы, нижеподписавшиеся составили акт о том, что следующие результаты диссертационной работы Бежаевой Оксаны Яковлевны, а именно:

- методология обеспечения функциональной надежности информационных систем на предпроектной стадии;
- комплекс структурных, контурных и динамических моделей проблемных ситуаций, связанных с обеспечением функциональной надежности информационных систем на предпроектной стадии;
- комплекс моделей ситуаций, возникающих при реализации программных проектов.

внедрены в учебный процесс кафедры Технической кибернетики и используются при изучении дисциплин: «Технико-экономическое обоснование программных проектов», «Проектирование и архитектура программных систем», «Разработка программных приложений», а также при разработке тем выпускных квалификационных работ бакалавров и магистров по направлению «Информатика и вычислительная техника»

Использование указанных материалов диссертационной работы познакомит студентов с современными подходами к обеспечению функциональной надежности информационных систем, позволит реализовать практические инструменты для моделирования ситуаций, возникающих при реализации проектов, обеспечит повышения качества процесса обучения.

Начальник управления  
развития образования

Гарипова Г.Т.

Директор института информатики,  
математики и робототехники

Кривошеева О.А.

## Приложение Б

(Детализация диаграмм раздела 6.1)

Детализации диаграммы Исикавы, построенной на основе выделенных кластеров запросов (рисунок 6.4), представлены на рисунках Б.1 – Б.17.

На рисунках Б.1 – Б.3 представлены диаграммы, детализации для кластера «Неверное представление данных в отчете» (рисунок 6.5)

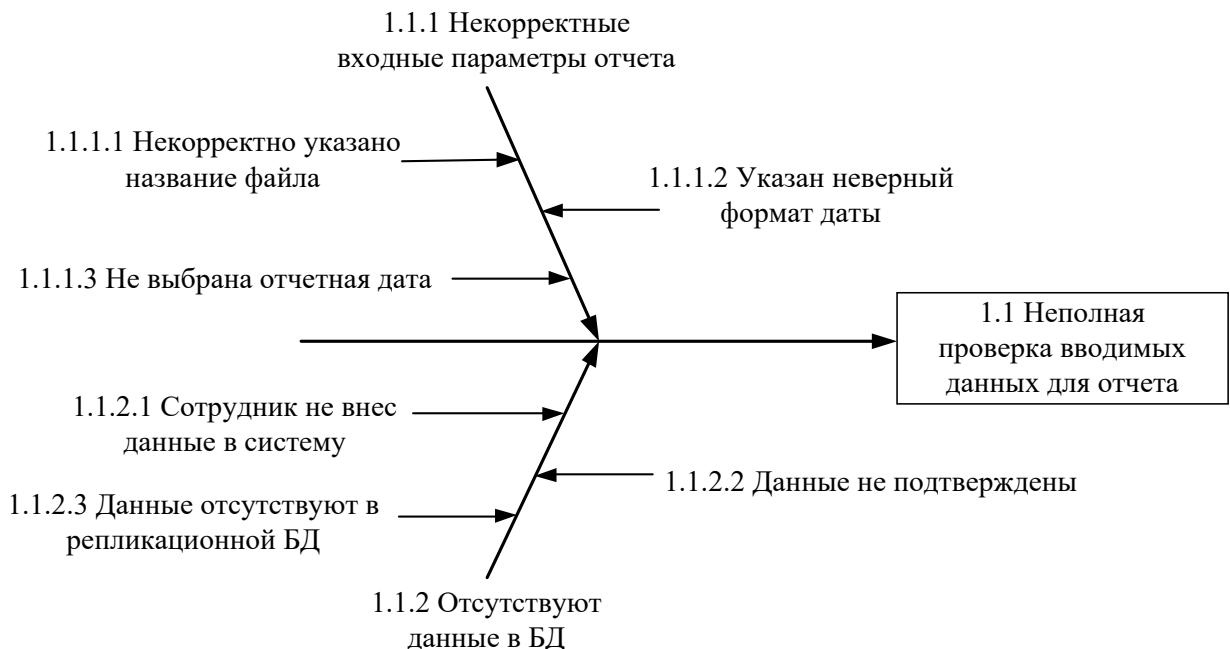


Рисунок Б.1 – Диаграмма Исикавы. Детализация для причины «Неполная проверка вводимых данных для отчета»

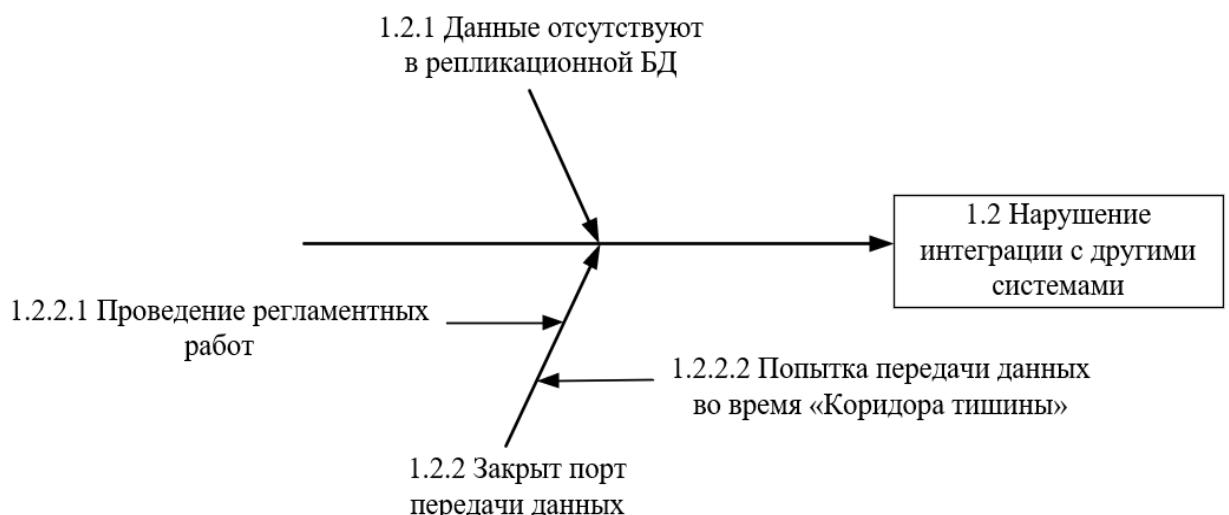


Рисунок Б.2 – Диаграмма Исикавы. Детализация для причины «Нарушение интеграции с другими системами»

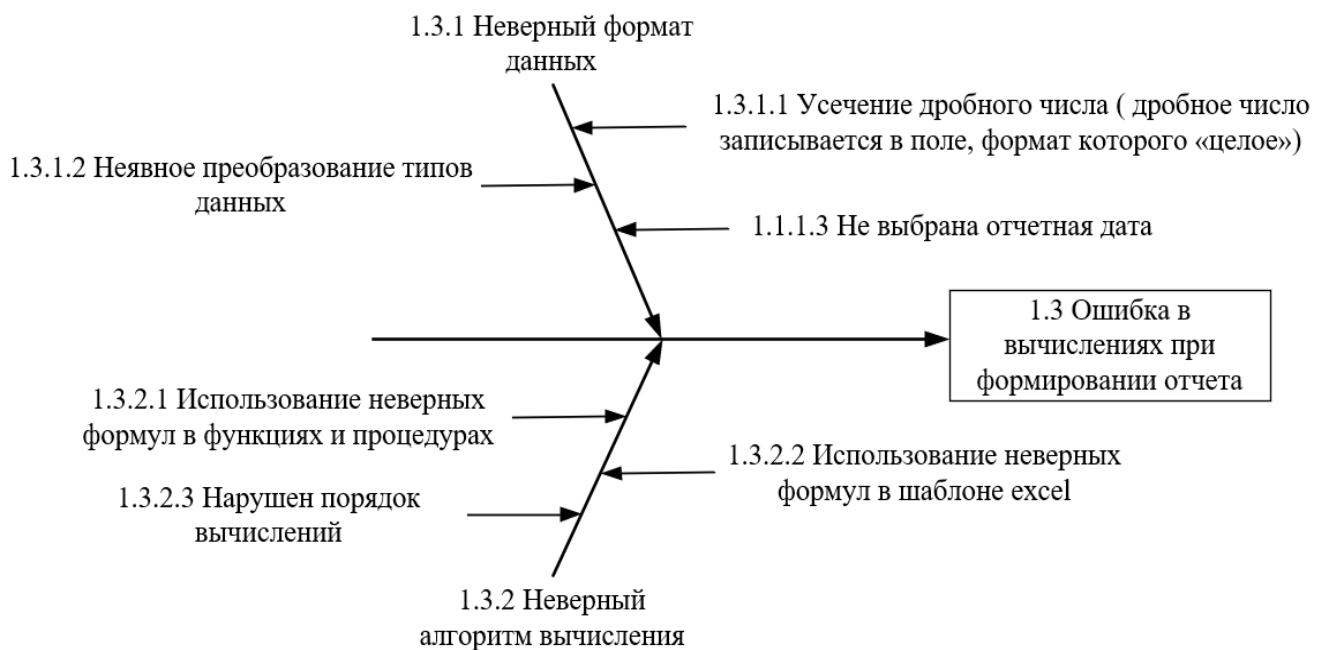


Рисунок Б.3 – Диаграмма Исикавы. Детализация для причины  
«Ошибка в вычислениях при формировании отчета»

На рисунке Б.4 представлена диаграмма Исикавы для причины «Модель не добавляется в систему». Как правило, пользователь встречается с данной проблемой, когда хочет добавить модель в модуле, отслеживать ее состояние и количественные показатели, но существует ряд причин, которые приводят к тому, что возникает исключительная ситуация при попытке добавить модель в систему.

Причинами данного следствия являются:

- неполная проверка вводимых данных;
- недостатки, связанные с прослеживанием доступа к каталогам;
- модель отсутствует в реестре;
- недостаточно полномочий учетных записей для регистрации модели.

Перечисленные выше причины в свою очередь являются следствием при возникновении других причин.

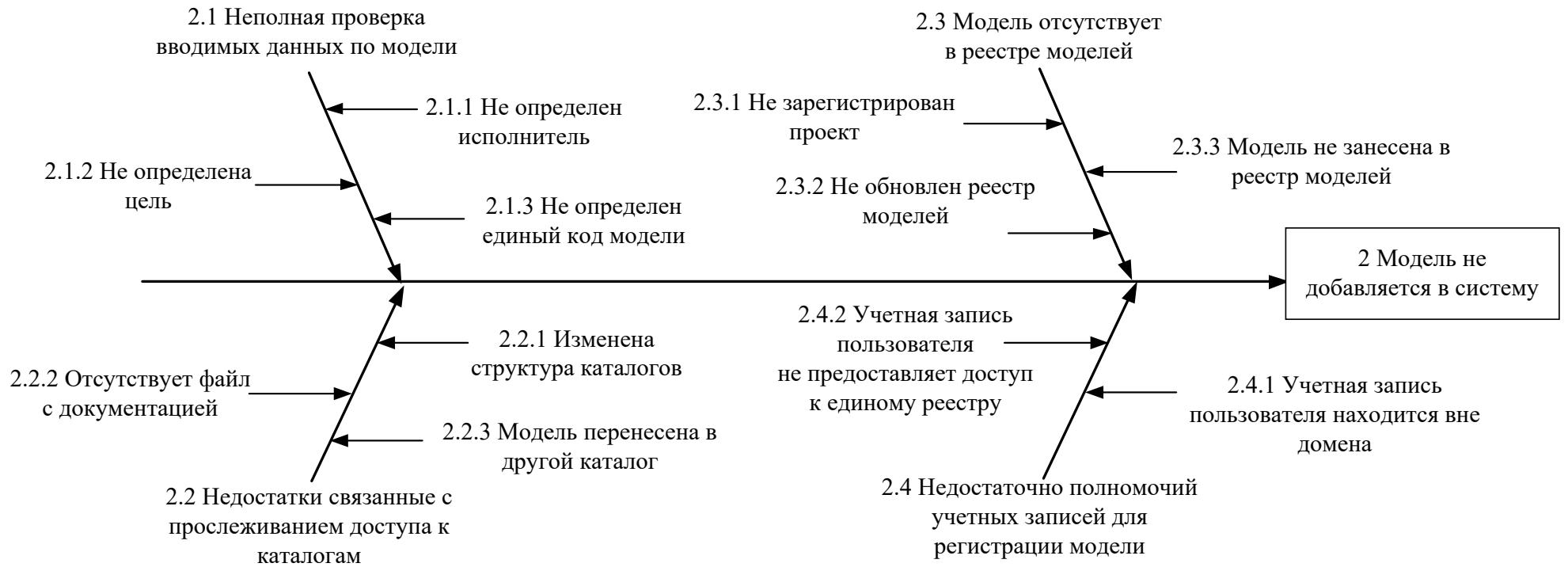


Рисунок Б.4 – Диаграмма Исикавы. Детализация для причины «Модель не добавляется в систему»

На рисунке Б.5 представлена диаграмма Исикавы для кластера «Не выгружается отчет из модуля», а рисунках Б.6 – Б.8 представлены детализации указанной диаграммы.

Пользователь встречается с данной проблемой при попытке выгрузить отчет из модуля. Как правило, в такой ситуации появляется ошибка о выгрузке отчета в формат Excel. Причинами возникновения данного следствия являются:

- некорректная настройка ПО;
- недостатки, связанные с прослеживанием каталогов;
- ошибка при формировании Excel файла.

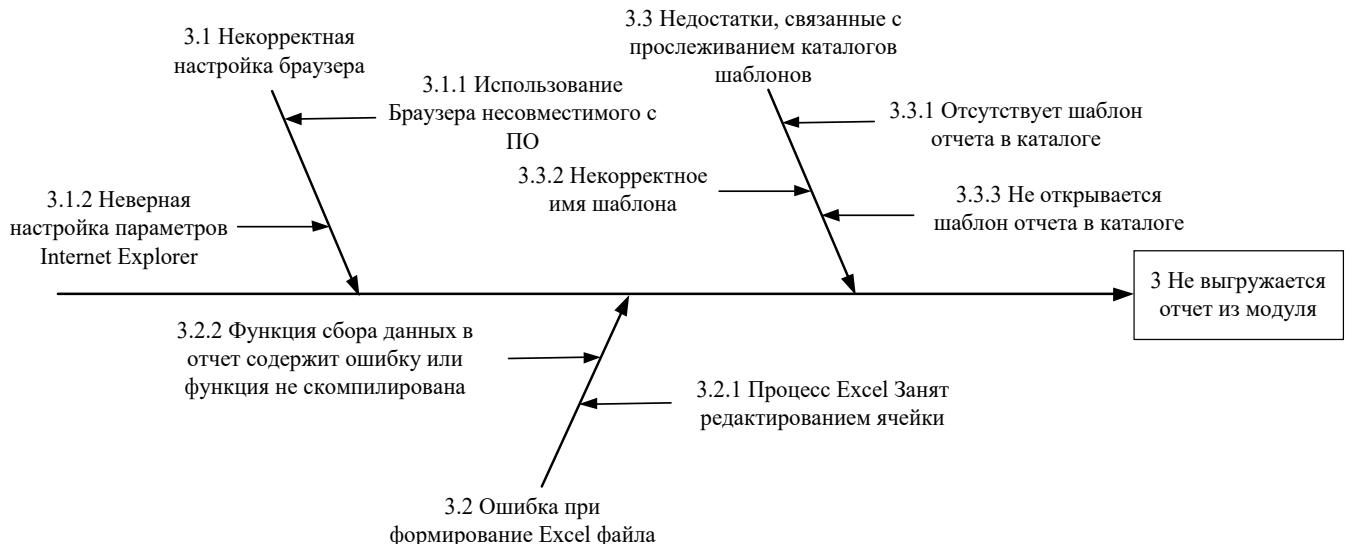


Рисунок Б.5 – Диаграмма Исикавы. Детализация для причины «Не выгружается отчет из модуля»

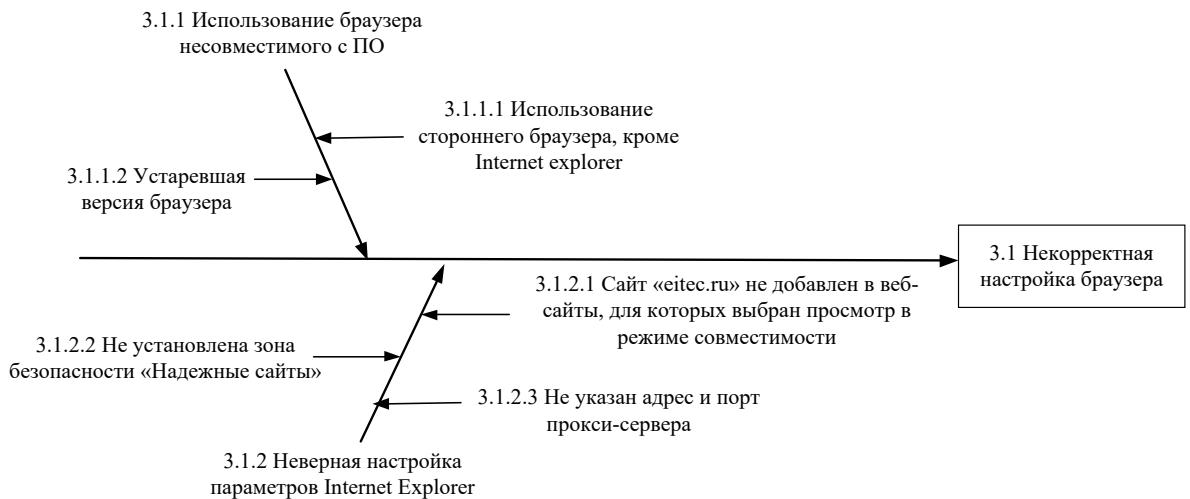


Рисунок Б.6 – Диаграмма Исикавы. Детализация для причины «Некорректная настройка браузера»

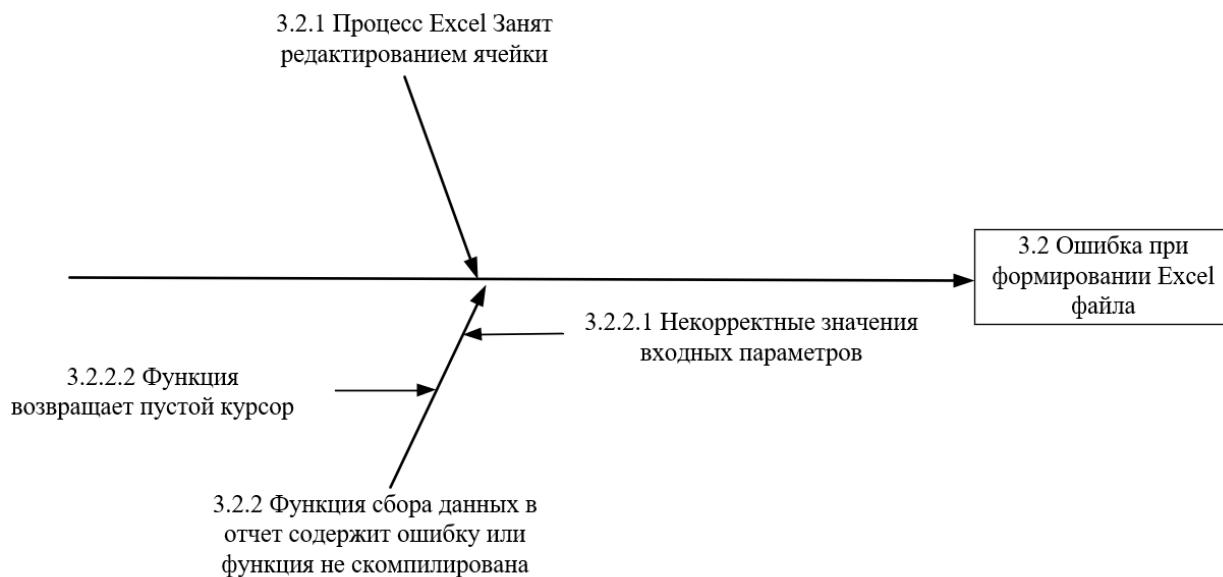


Рисунок Б.7 – Диаграмма Исикавы. Детализация для причины «Ошибка при формировании Excel файла»



Рисунок Б.8 – Диаграмма Исикавы. Детализация для причины «Недостатки, связанные с прослеживанием каталогов шаблона»

На рисунке Б.9 представлена диаграмма Исикавы для кластера «Не сохраняются данные в БД», а рисунках Б.10 – Б.11 представлены детализации указанной диаграммы. Пользователь встречается с данной проблемой при введении данных через форму ввода.

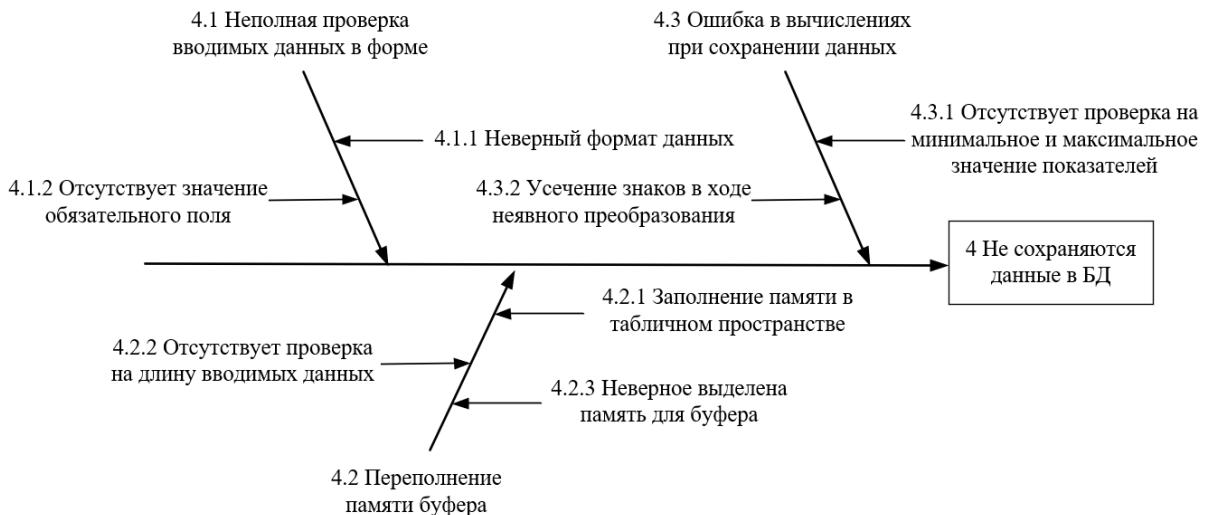


Рисунок Б.9 – Детализация для причины «Не сохраняются данные в БД»

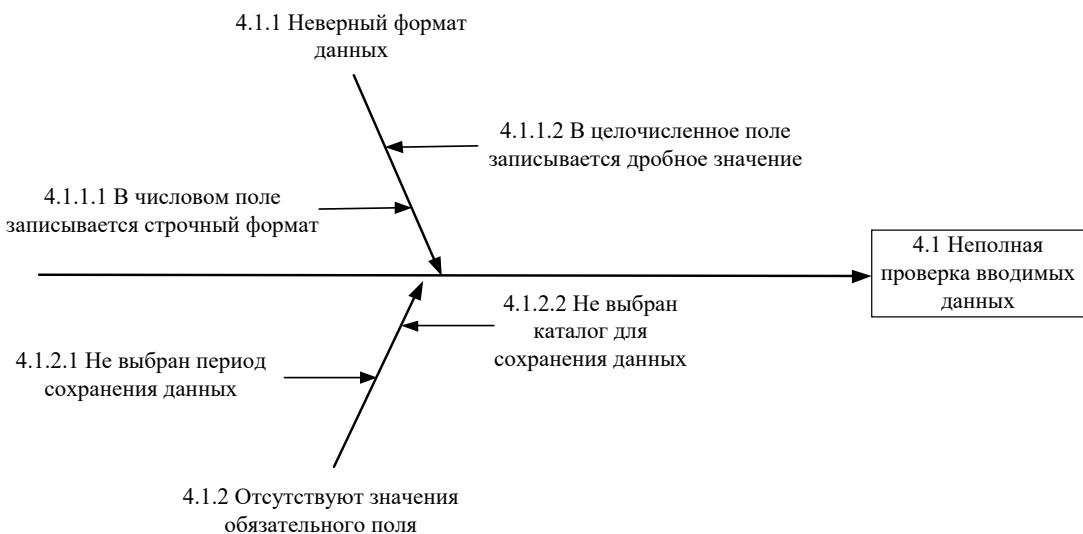


Рисунок Б.10 – Детализация для причины «Неполная проверка вводимых данных»

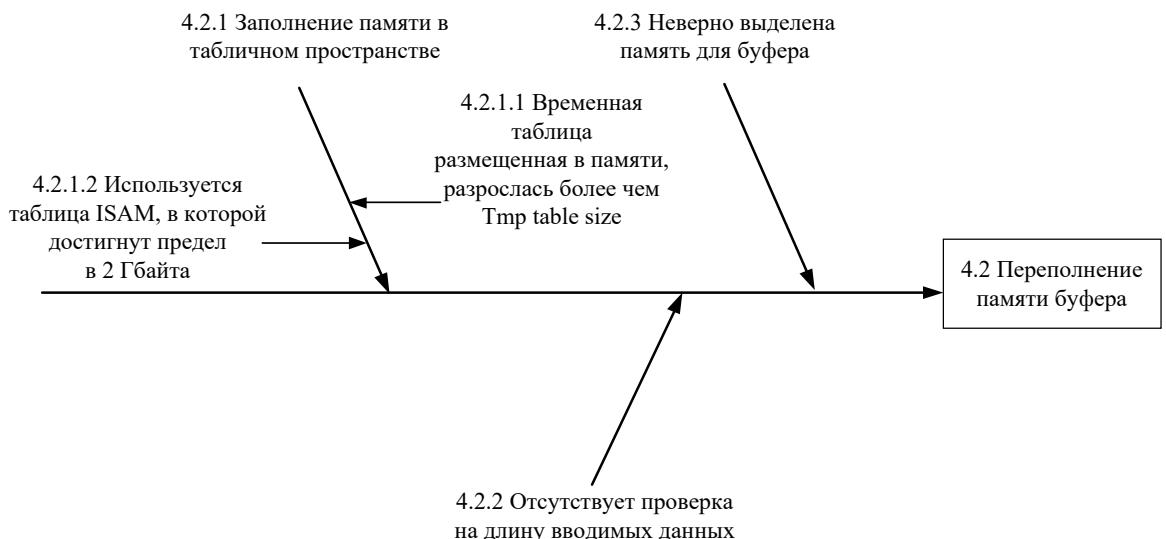


Рисунок Б.11 – Детализация для причины «Переполнение памяти буфера»

На рисунке Б.12 представлена диаграмма Исикиавы для кластера «Не загружаются данные из файла». Пользователь встречается с данной проблемой при введении данных через форму ввода, используя заполненный Excel файл. Причинами возникновения данного следствия являются:

- неполная проверка вводимых данных в форме;
- переполнение памяти буфера;
- ошибка в вычислениях при сохранении данных;
- недостатки, связанные с прослеживанием каталогов шаблонов форм.



Рисунок Б.12 – Диаграмма Исикиавы. Детализация для причины «Не загружаются данные из файла»

На рисунке Б.13 представлена диаграмма Исикиавы для кластера «Не загружаются данные из файла».

Пользователь встречается с данной проблемой при попытке входа в систему. По ряду причин пользователь терпит неудачи при попытке авторизации. На рисунке Б.14 представлена детализация для причины «Некорректная настройка браузера». указанной диаграммы.

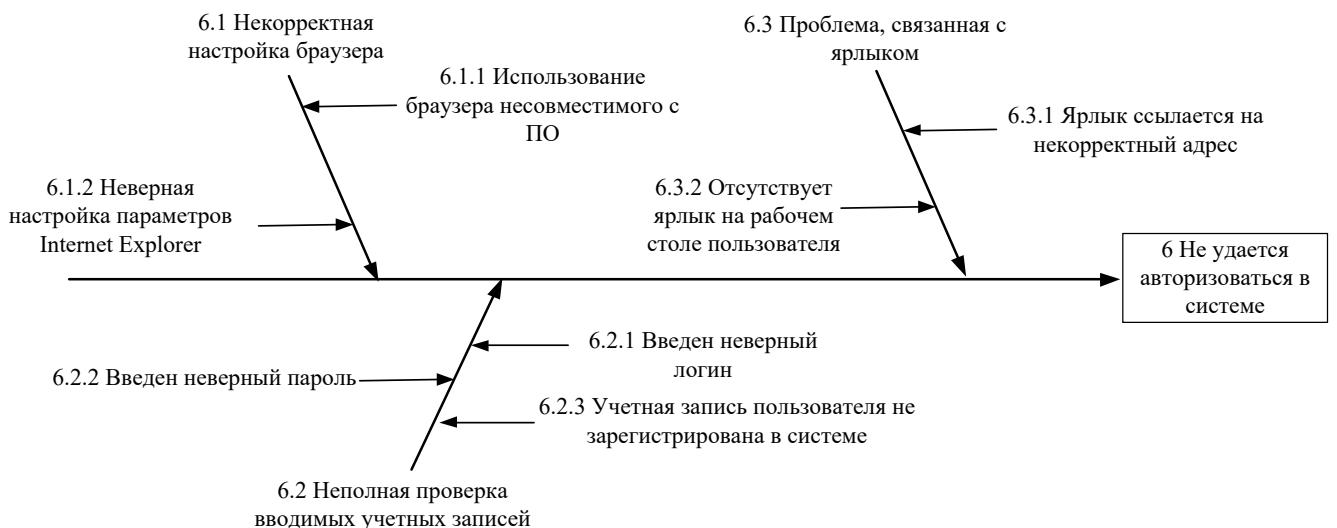


Рисунок Б.13 – Диаграмма Исикавы. Детализация для причины «Не удается авторизоваться в системе»

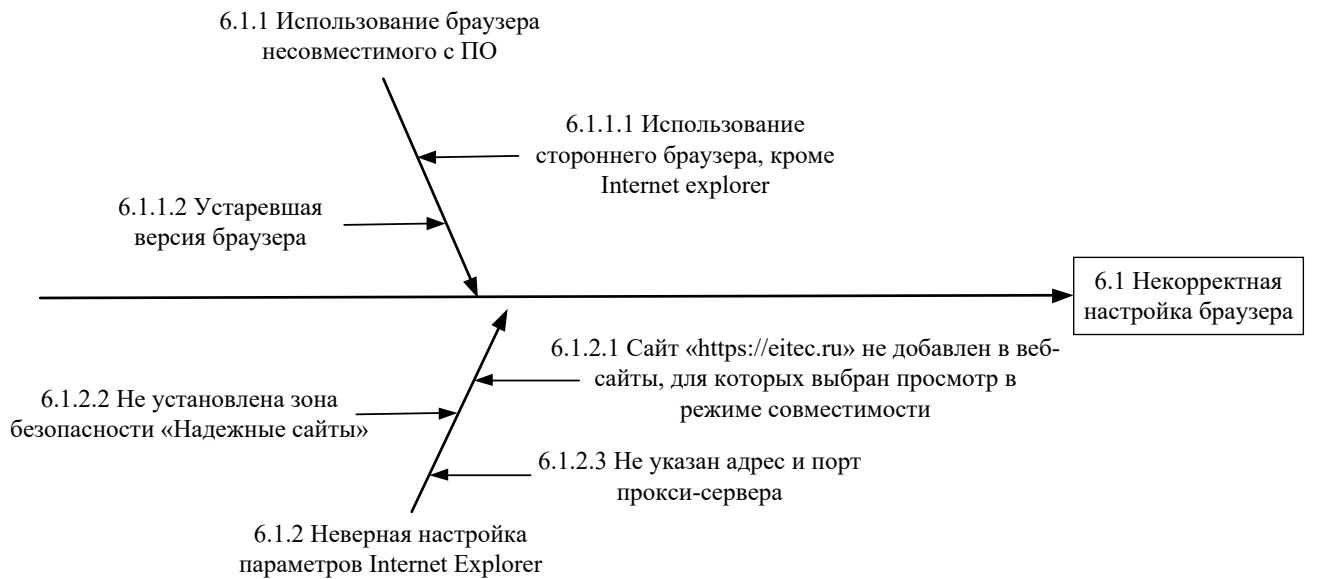


Рисунок Б.14 – Диаграмма Исикавы. Детализация для причины «Некорректная настройка браузера»

На рисунке Б.15 представлена диаграмма Исикавы для кластера «Сбой рассылки отчетов», на рисунках Б.17 – Б.18 детализации указанной диаграммы.

Некоторые пользователи системы в определенное время получают на почту автоматически сформированный отчет из системы, возникают причины, при которых пользователь не получает отчет на почту. Причинами могут быть:

- переполнение памяти на сервере;
- неполадки с сетью;
- некорректная настройка ПО.

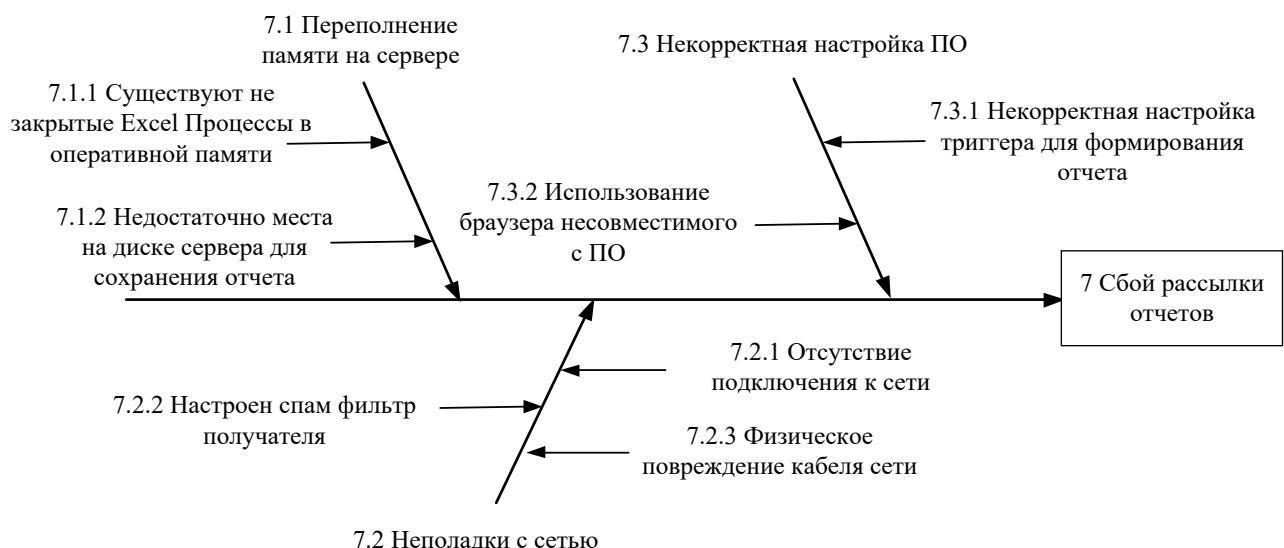


Рисунок Б.15 –Детализация для причины «Сбой рассылки отчетов»

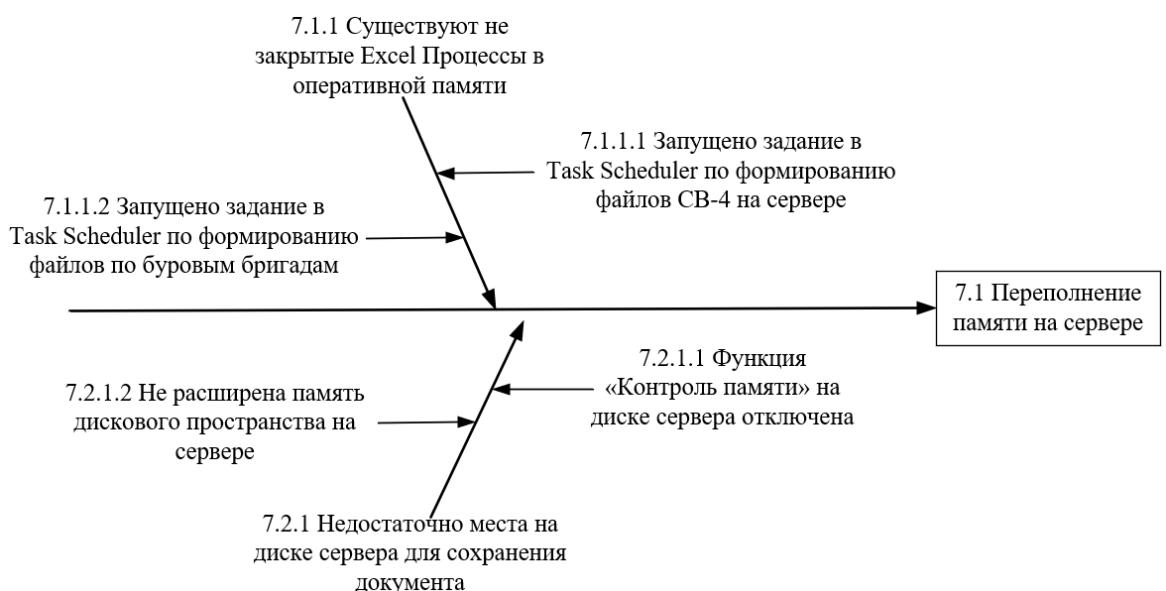


Рисунок Б.16 –Детализация для причины «Переполнение памяти сервера»

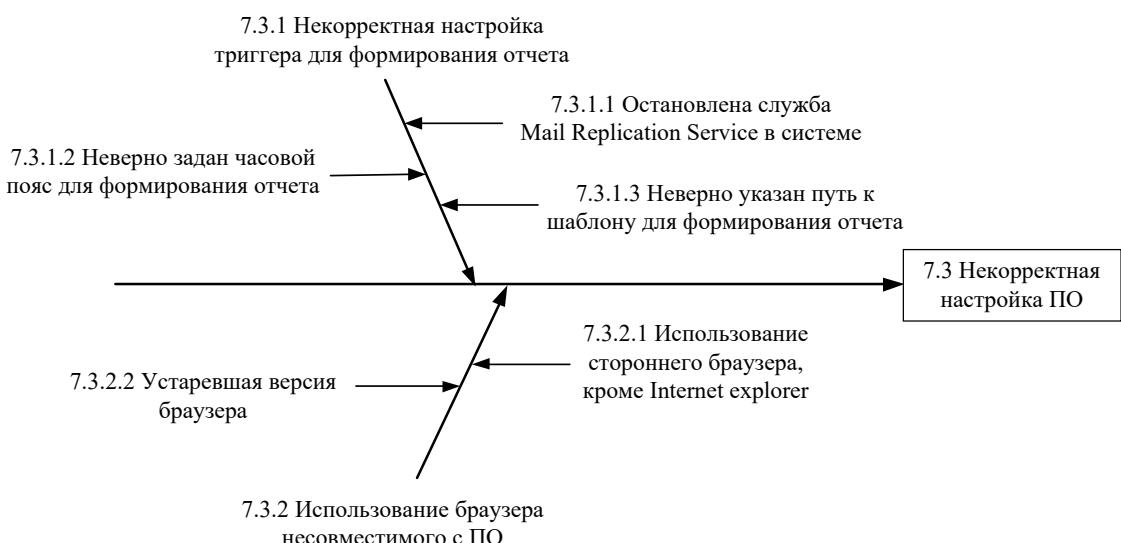


Рисунок Б.17 – Детализация для причины «Некорректная настройка ПО»

На следующих этапах анализа функциональной надежности используется метод анализа дерева отказов (англ. *FTA*). На рисунке 6.6 показана схема первого уровня дерева отказов, декомпозиции представлены на рисунках Б.18-Б.25.

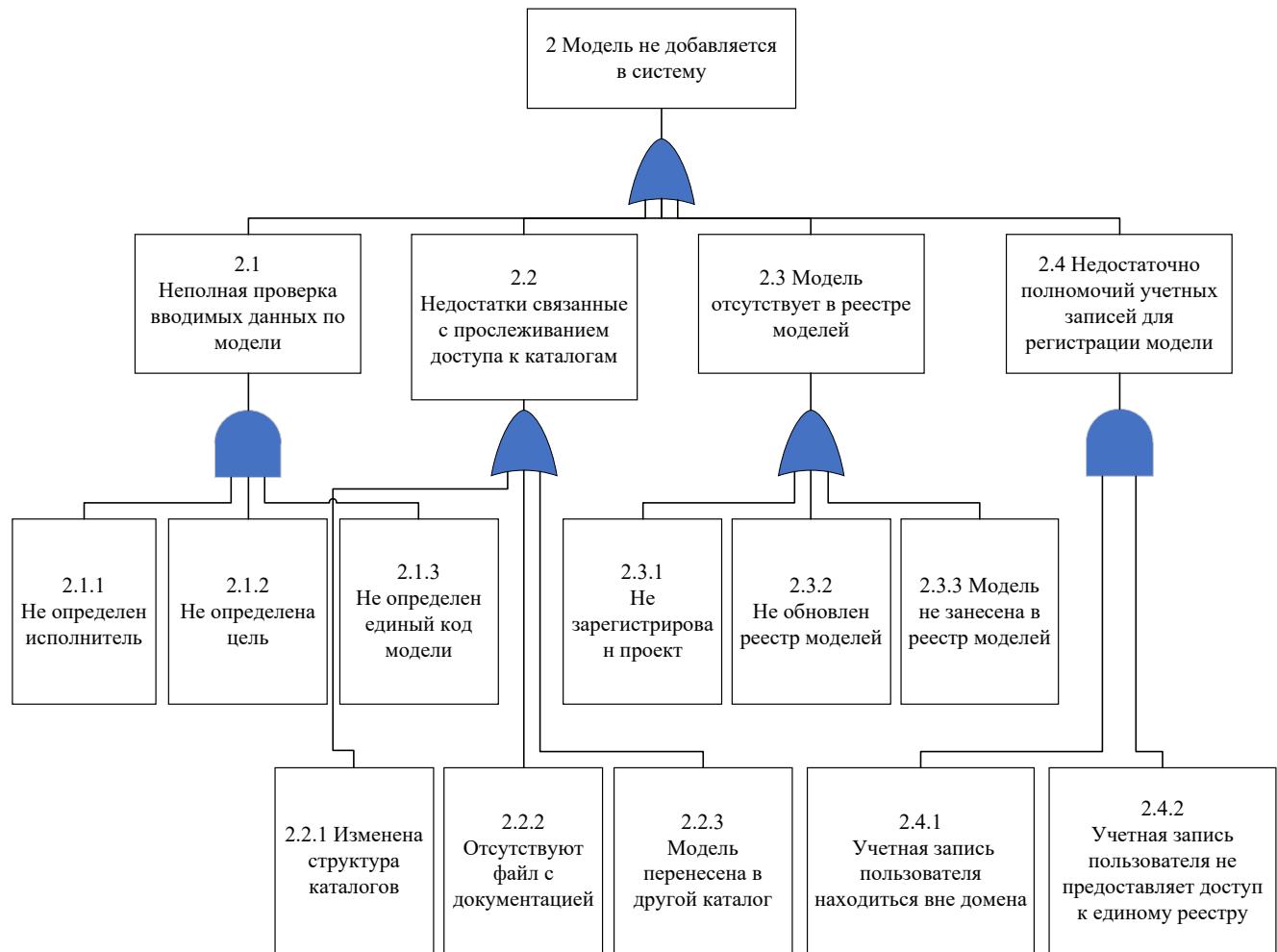


Рисунок Б.18 – FTA. Модель не добавляется в систему

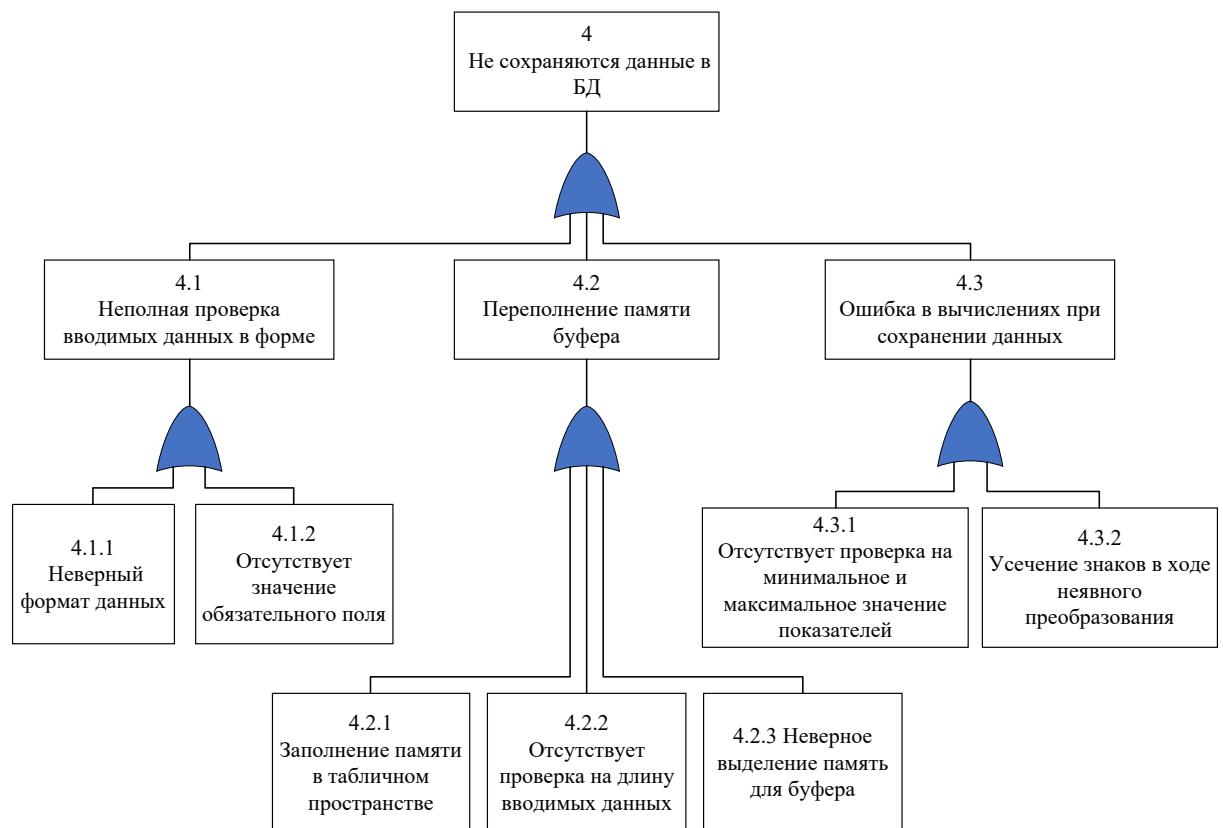
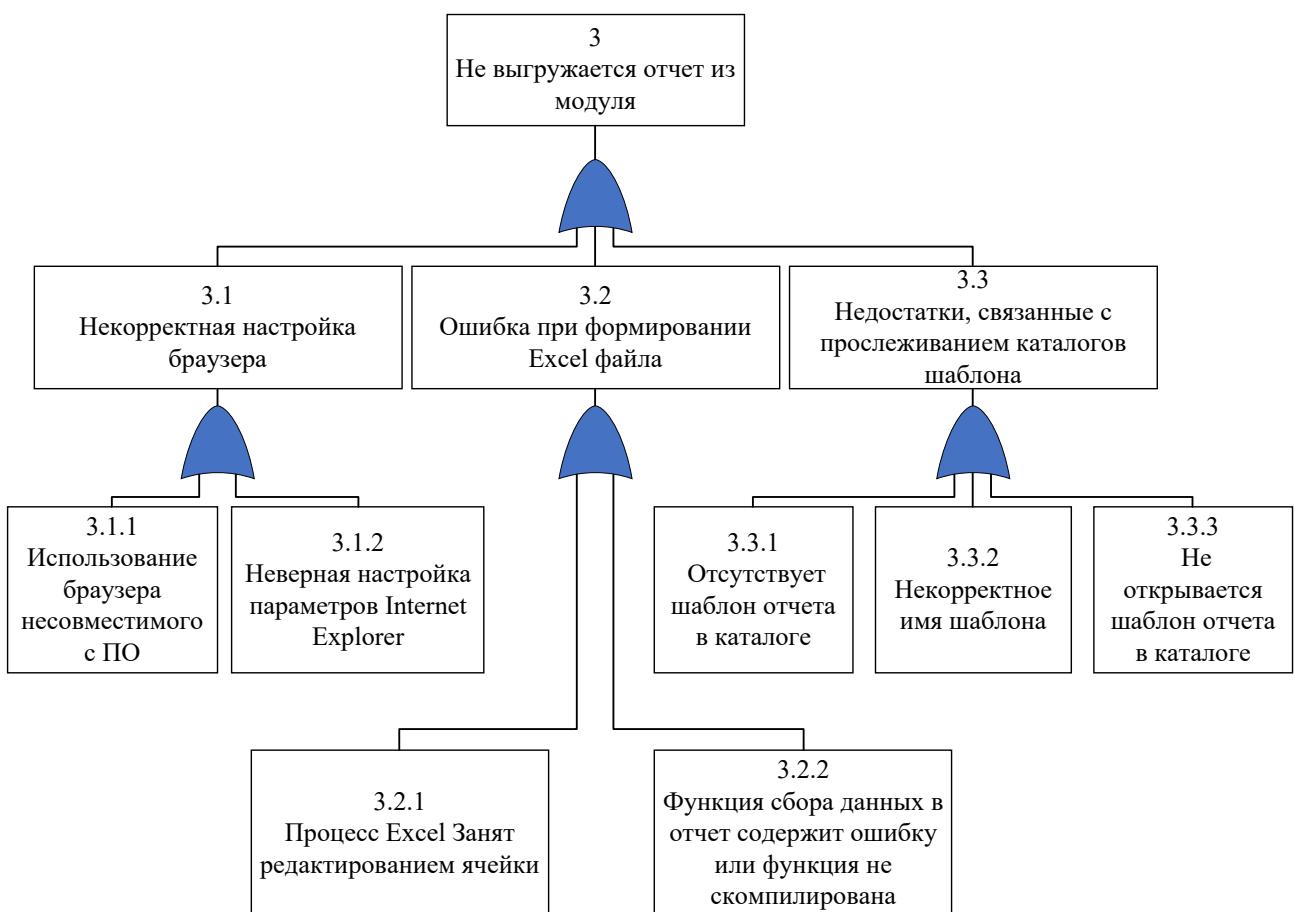


Рисунок B.20 – FTA. Не сохраняются данные в БД

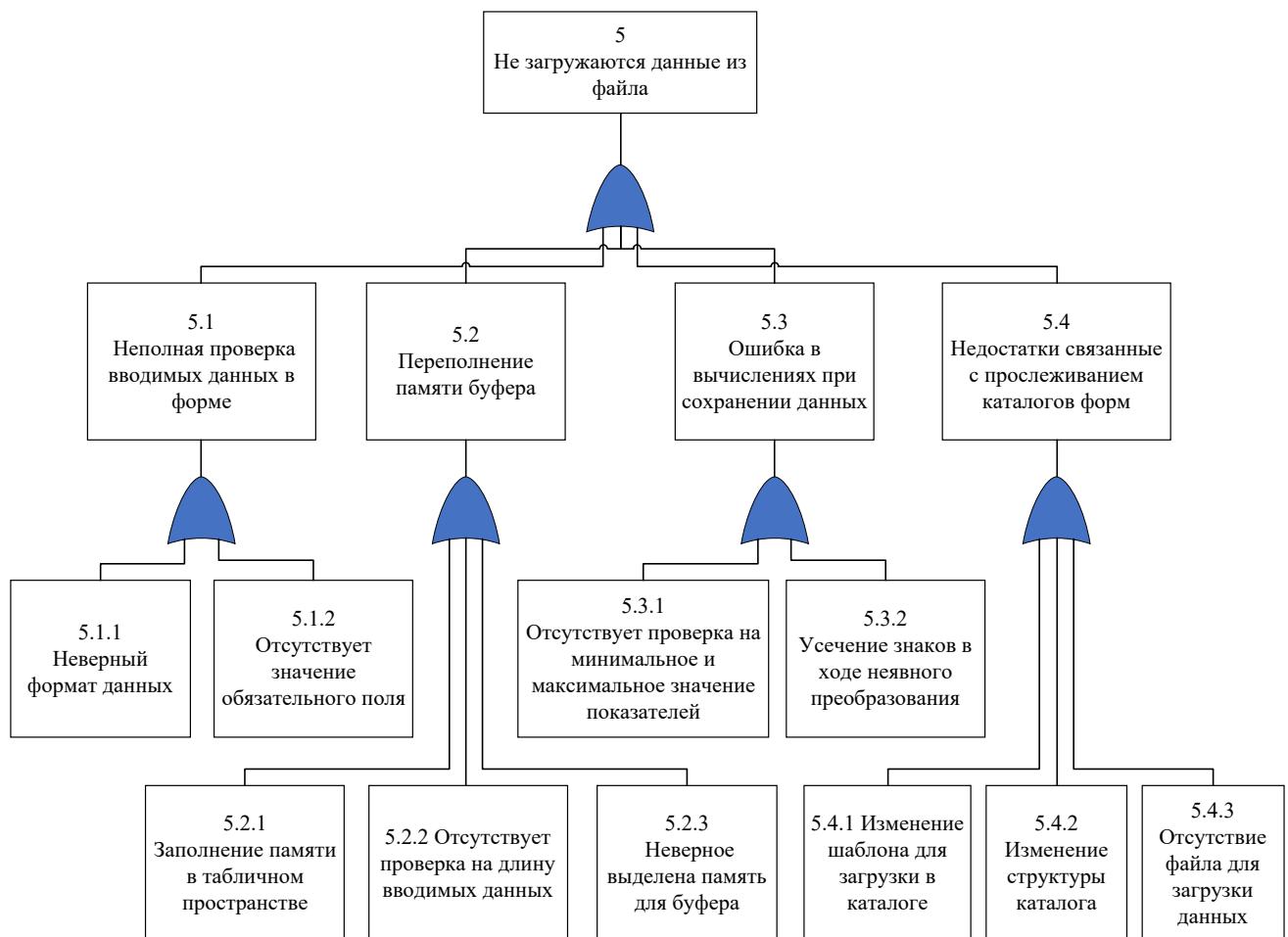


Рисунок Б.21 – FTA. Не загружаются данные из файла

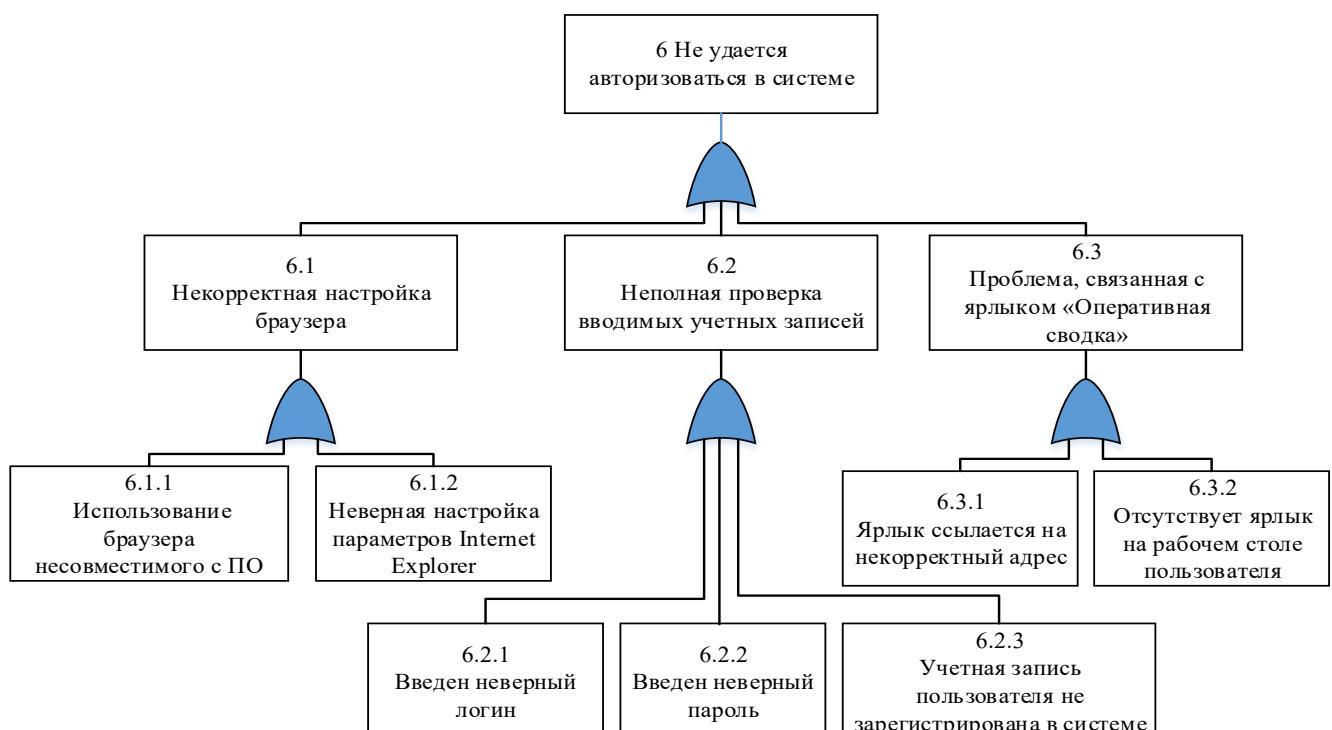


Рисунок Б.22 – FTA. Не удается авторизоваться в системе

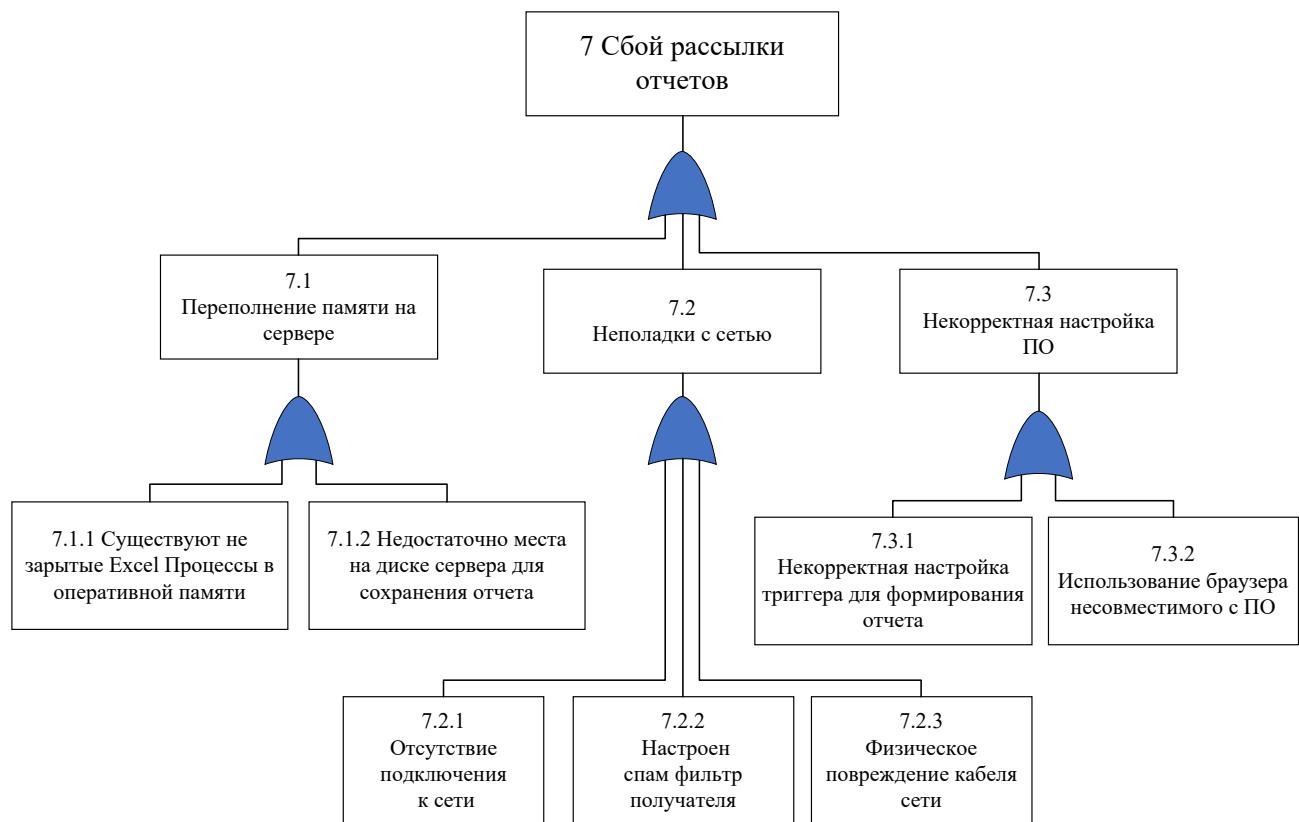


Рисунок Б.23 – FTA. Сбой рассылки отчетов

## Приложение В

(Диаграммы UML раздела 6.2)

Диаграмма вариантов использования (*Use Case*) представлена на рисунке В-1.

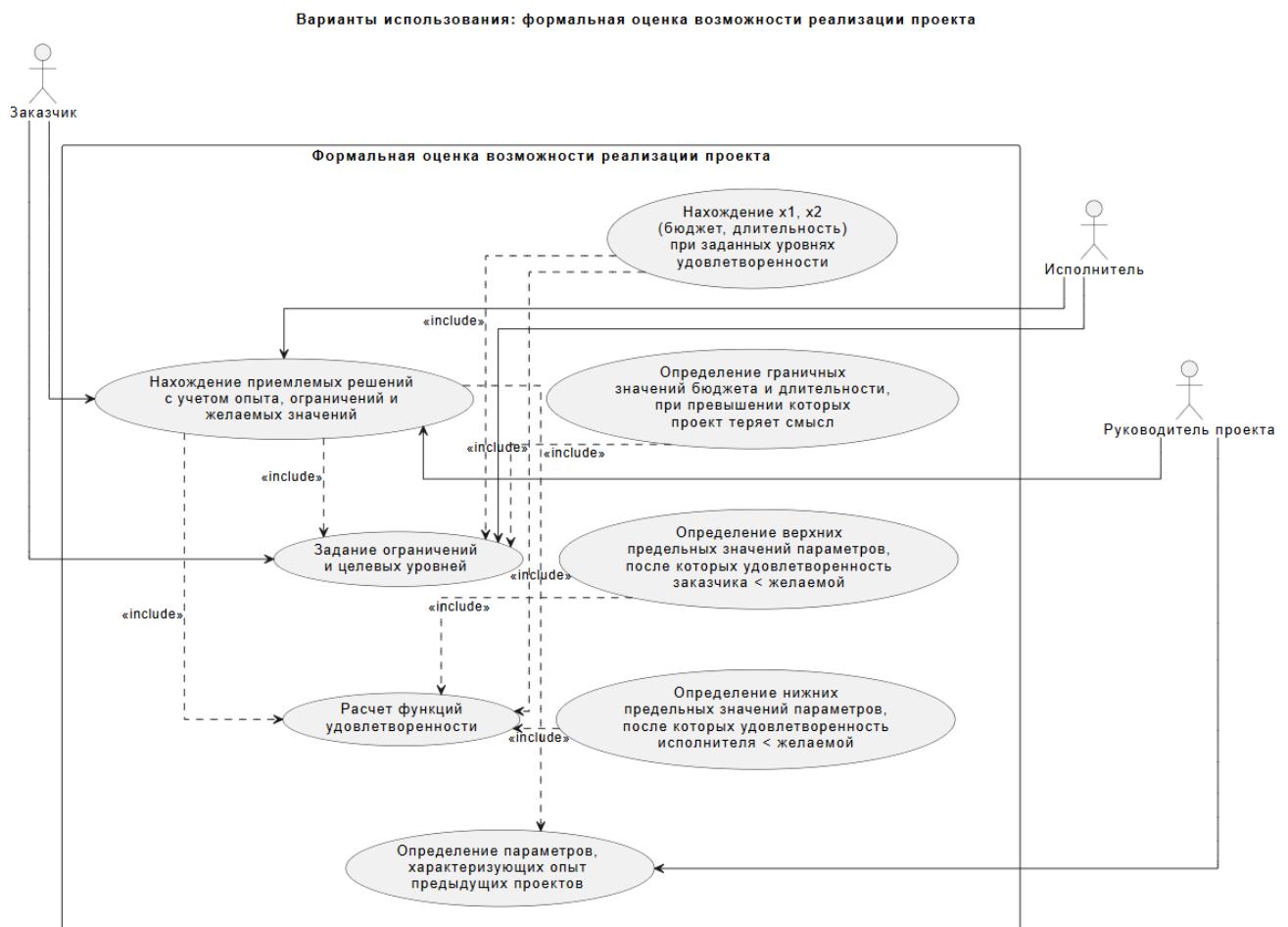


Рисунок В.1 – Диаграмма вариантов использования (*Use Case*) –  
Формальная оценка возможности реализации проекта

В таблицах В.1 – В.4 представлено детальное описание диаграммы вариантов использования (*Use Case*).

**Таблица В.1 – Акторы**

Сущность	Роль
Заказчик	Задает целевые уровни удовлетворенности, ограничения и желаемые параметры проекта.
Исполнитель	Участвует в расчётах, задаёт собственные целевые уровни удовлетворенности.
Руководитель проекта	Координирует процесс формальной оценки, получает итоговые результаты.

Таблица В.2 - Варианты использования верхнего уровня

Вариант использования	Назначение	Основные результаты	Задействованные подфункции
Формальная оценка возможности реализации проекта	Комплексная проверка достижимости целей проекта при заданных ограничениях и опыте.	Заключение о возможности реализации, отчет с результатами, диапазоны параметров.	Нахождение приемлемых решений; Задание ограничений и целевых уровней; Расчет функций удовлетворенности; Определение параметров опыта.
Нахождение приемлемых решений	Подбор параметров бюджета и сроков, удовлетворяющих условиям и целевым уровням.	Набор допустимых решений ( $x_1, x_2$ ), учитывающих ограничения и опыт.	Нахождение $x_1, x_2$ ; Определение граничных значений бюджета и длительности.
Задание ограничений и целевых уровней	Формализация исходных ограничений и целевых уровней удовлетворенности.	Система ограничений для анализа.	Определение верхних и нижних предельных значений параметров.
Расчет функций удовлетворенности	Построение функций $Sat_c$ и $Sat_v$ на основе опыта и ограничений.	Количественные оценки удовлетворенности заказчика и исполнителя.	Использование данных опыта и моделей удовлетворенности.
Определение параметров опыта предыдущих проектов	Извлечение и анализ данных прошлых проектов для использования в расчетах.	Параметры, характеризующие опыт, применяемые в дальнейших вычислениях.	Сбор и формализация данных из истории проектов.

Таблица В.3 - Детализация подфункций

Базовый вариант	Отношение	Подфункция	Краткое описание результата
Нахождение приемлемых решений	include	Нахождение $x_1, x_2$ (бюджет, длительность)	Определение конкретных значений параметров при заданных уровнях удовлетворенности.
Нахождение приемлемых решений	include	Определение граничных значений бюджета и длительности	Фильтрация решений, при которых проект теряет смысл.
Задание ограничений и целевых уровней	include	Определение верхних предельных значений параметров	Выявление пределов, после которых удовлетворенность заказчика меньше требуемой.
Задание ограничений и целевых уровней	include	Определение нижних предельных значений параметров	Выявление пределов, после которых удовлетворенность исполнителя меньше требуемой.
Расчет функций удовлетворенности	include	Определение параметров, характеризующих опыт предыдущих проектов	Формирование базы параметров для расчетов функций удовлетворенности.

Таблица В.4 - Триггеры, входы и выходы

Вариант использования	Триггер	Входные данные	Выходные данные
Формальная оценка возможности реализации проекта	Поступает запрос на проверку возможности реализации проекта	Запрос, ограничения, опыт предыдущих проектов	Итоговое заключение о возможности реализации проекта
Нахождение приемлемых решений	Определены исходные условия	Опыт, ограничения, целевые уровни	Набор приемлемых решений
Задание ограничений и целевых уровней	Заказчик/исполнитель вводят параметры	Бюджет, сроки, желаемые уровни удовлетворенности	Система ограничений для анализа
Расчет функций удовлетворенности	Заданы исходные параметры	Модели удовлетворенности, данные об ограничениях	Количественные значения удовлетворенности
Определение параметров опыта предыдущих проектов	Доступны данные прошлых проектов	История проектов	Параметры, используемые в расчётах

Пояснения к диаграмме (рисунок В.1):

1) Запрос на формальную оценку и ввод целевых уровней удовлетворенности запускают выполнение базовых вариантов использования. Действия заказчика и исполнителя формируют исходные данные, которые система обрабатывает для получения результатов.

2) Отношение *include* означает обязательное выполнение подфункции в составе базового варианта использования. Например, «Нахождение приемлемых решений» обязательно включает «Нахождение x1, x2 (бюджет, длительность)» и «Определение граничных значений бюджета и длительности». Аналогично, «Задание ограничений» включает определение верхних и нижних предельных значений параметров.

3) Структура диаграммы задает логику перехода от вводных к результатам: участники задают цели и ограничения; система извлекает параметры опыта прошлых проектов; затем выполняются обязательные подфункции, формируется множество допустимых решений и выделяются итоговые рекомендации.

4) Типовой сценарий выполнения: актор инициирует процедуру оценки; система загружает опыт и ограничения; выполняются подфункции, помеченные include; рассчитывается уровень удовлетворенности сторон и определяются пороги; на их основе формируется область допустимых решений, а также рекомендации о возможности/невозможности нахождения приемлемых решений.

Диаграмма вариантов использования (*Use Case*) представлена на рисунке В-2. В таблицах В.5 – В.7 представлено подробное описание диаграммы.

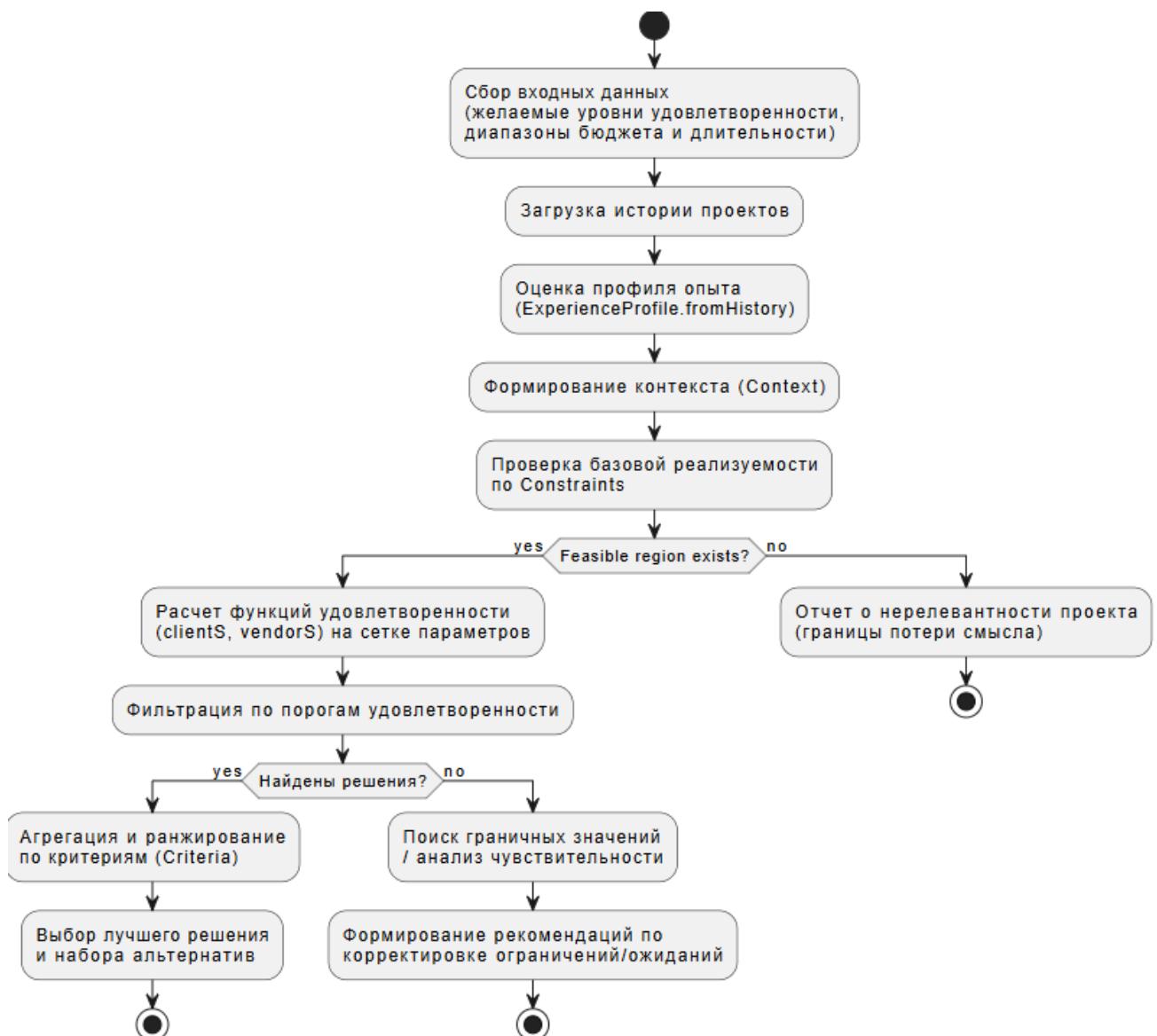


Рисунок В.2 – Диаграмма активности (Activity Diagram) – Формальная оценка возможности реализации проекта

Диаграмма описывает поток работ от получения исходных данных до выдачи рекомендаций или отчета. Процесс состоит из трех фаз:

- подготовка данных (A1–A4);
- проверка реализуемости (D1);
- аналитика (A6–A11).

Если допустимая область отсутствует, формируется отчет о нерелевантности (A5→F1). При существовании допустимой области вычисляются функции удовлетворенности (A6), проводится фильтрация (A7), и проверяется наличие решений (D2). Если решения есть, они агрегируются и ранжируются (A8), затем выбирается лучшее решение и альтернативы (A9→F2). Если решений нет, система выполняет анализ чувствительности (A10) и формирует рекомендации по корректировке ожиданий (A11→F3).

Таблица В.5 - Узлы диаграммы активностей

ID	Тип узла	Наименование	Назначение / результат	Входы	Выходы
N0	Initial	Старт процесса	Инициализация сценария	-	-
A1	Action	Сбор входных данных	Получение целевых уровней удовлетворенности, диапазонов бюджета и длительности	Запрос на оценку	Вводные параметры
A2	Action	Загрузка истории проектов	Подготовка эмпирической базы для калибровок	Вводные параметры	Данные истории
A3	Action	Оценка профиля опыта	Построение профиля опыта для последующих расчетов	Данные истории	Профиль опыта
A4	Action	Формирование контекста	Композиция вводных, профиля опыта и ограничений	Вводные параметры; Профиль опыта	Контекст
D1	Decision	Проверка реализуемости	Определение существования допустимой области	Контекст	yes/no

ID	Тип узла	Наименование	Назначение / результат	Входы	Выходы
A5	Action	Отчет о нерелевантности проекта	Формирование отчета	Контекст	Отчет о нерелевантности
F1	Final	Завершение (ветка нет)	Завершение процесса при отсутствии допустимой области	Отчет о нерелевантности	—
A6	Action	Расчет функций удовлетворенности	Вычисление clientS и vendorS	Контекст; Сетка параметров	Уровень удовлетворенности
A9	Action	Выбор решения и альтернатив	Формирование рекомендации	Ранжированный список	Рекомендованное решение; Альтернативы
F2	Final	Завершение (ветка да)	Завершение успешного сценария	Рекомендованное решение; Альтернативы	—
A10	Action	Поиск граничных значений	Определение ограничений	Контекст	Границные значения;
A11	Action	Формирование рекомендаций по корректировке	Предложения по изменению ограничений/ожиданий	Границые значения;	Рекомендации
F3	Final	Завершение	Завершение сценария с рекомендациями	Рекомендации	—

Таблица В.6 - Таблица — Переходы и условия

Откуда → Куда	Условие/метка	Комментарий
N0 → A1	—	Инициация процесса
A1 → A2	—	Ввод данных → загрузка истории
A2 → A3	—	История используется для построения профиля опыта
A3 → A4	—	Формирование контекста
A4 → D1	—	Проверка допустимости
D1 → A6	yes	Допустимая область существует
D1 → A5	no	Допустимая область отсутствует
A5 → F1	—	Формируется отчет о нерелевантности
A6 → A7	—	Расчет → фильтрация
A7 → D2	—	Проверка непустоты множества
D2 → A8	yes	Есть решения
D2 → A10	no	Решений нет
A8 → A9	—	Формирование итоговой рекомендации
A9 → F2	—	Завершение успешной ветки
A10 → A11	—	Определение граничных значений
A11 → F3	—	Формирование рекомендаций по корректировке

Таблица В.7 - Артефакты данных

Артефакт	Источник	Используется в	Описание
Вводные параметры	A1	A4, D1, A6, A7	Цели, диапазоны бюджета и длительности, ограничения
Данные истории	A2	A3	Архив сведений о прошлых проектах
Профиль опыта	A3	A4, A6	Агрегированные параметры и эмпирические диапазоны
Контекст	A4	D1, A6, A10	Композиция вводных и профиля опыта
Границочные значения	A10	A11	Активные ограничения
Рекомендации по корректировке	A11	F3	Предложения по изменению ограничений/ожиданий
Отчет о нерелевантности	A5	F1	Фиксирует невозможность реализации проекта

Диаграмма последовательностей (Sequence Diagram) представлена на рисунке В-3. Диаграмма иллюстрирует согласованный процесс от постановки целей до выдачи рекомендаций и отчетов.

В таблицах В.8 – В.9 представлено подробное описание диаграммы.

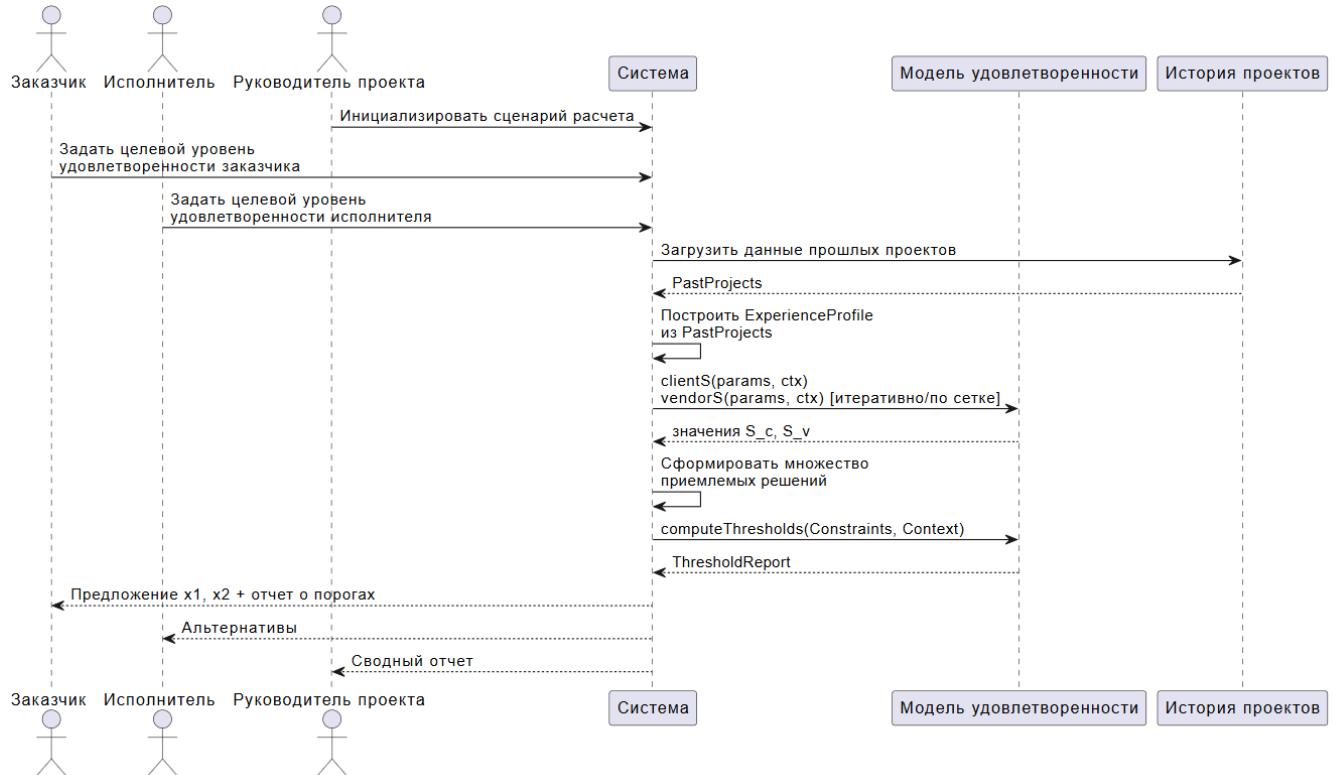


Рисунок В.3 – Диаграмма последовательностей (Sequence Diagram) –  
Формальная оценка возможности реализации проекта

Таблица В.8 - Участники взаимодействия

Обозначение	Роль	Краткое описание ответственности
C	Заказчик	Задает целевой уровень удовлетворенности заказчика, получает предложение параметров $x_1$ , $x_2$ и отчет о порогах.
V	Исполнитель	Задает целевой уровень удовлетворенности исполнителя, получает набор альтернатив и комментарии по рискам.
PM	Руководитель проекта	Инициирует сценарий расчета, получает сводный отчет и утверждает результаты.
S	Система	Оркеструет процесс: загружает историю, строит профиль опыта и контекст, считает уровень удовлетворенности, формирует решения и отчеты.
Sat	Модель удовлетворенности	Вычисляет значения функций удовлетворенности (clientS, vendorS), пороги и связанные отчеты.
Hist	История проектов	Источник данных о прошлых проектах (PastProjects) для построения ExperienceProfile.

Таблица В.9 - Данные и артефакты

Артефакт	Источник	Потребители	Назначение
PastProjects	Hist	S	Исторические данные для построения профиля опыта
ExperienceProfile	S	S	Сводные параметры опыта.
Constraints	C, V	S, Sat	Ограничения по бюджету и срокам
Context	S	S, Sat	Композиция входных, профиля опыта и ограничений
Grid(params)	S	Sat	Сетка параметров ( $x_1$ , $x_2$ )
S_c, S_v	Sat	S	Значения удовлетворенности на сетке
FeasibleSolutions	S	S	Допустимые решения
ThresholdReport	Sat	S, C, V, PM	Границы потери смысла
Alternatives	S	V	Альтернативные варианты
SummaryReport	S	PM	Сводный отчет для руководителя проекта.