Министерство образования и науки Российской Федерации
ФГБОУ ВПО «Байкальский государственный университет экономики и права»

На правах рукописи

Артамонов Иван Васильевич

Исследование надежности бизнес-транзакций в сервис-ориентированной среде

Специальность 05.13.01 – Системный анализ, управление и обработка информации

Диссертация на соискание ученой степени кандидата технических наук

Научный руководитель
д.э.н., профессор Н.В. Амбросов

Иркутск – 2015

Оглавление

Введение

В современном мире возрастает роль и значимость взаимодействий на основе информационных технологий. В секторе экономических отношений последние десятилетия кооперация с помощью электронных систем стала иметь ведущее значение. Электронное взаимодействие с поставщиками и потребителями, государственными органами и разными посредниками позволяет существенно ускорить темпы исполнения бизнес-процессов, увеличить скорость производства и продаж, уменьшить количество ошибок и повысить качество работы.

Несколько предприятий могут одновременно реализовывать общий бизнес-процесс или являться исполнителями процесса, инициированного одной стороной, результаты которого используются другой. Такой процесс пересекает границы организаций, находится в области ответственности нескольких участников и может длиться долгое время.

Вместе с тем немаловажное значение принимает интеграция и внутри самого предприятия. Современное состояние рынка корпоративных программных средств позволило оснастить разные отделы, цеха, территориально-распределенные департаменты и отдельные уровни управления собственной номенклатурой аппаратно-программных средств. Нередки ситуации, когда деятельность компании автоматизирована несколькими разнородными системами разных поставщиков, которые вынуждены кооперироваться друг с другом. Поэтому последние 20 лет остро стоит проблема интеграции корпоративных приложений друг с другом и поддержки унаследованных, устаревших, но все еще работоспособных систем. Одним из решений проблем интеграции стала сервис-ориентированная архитектура, идеология и стандарты которой разрабатываются международным независимым консорциумом.

В стремлении к повсеместной автоматизации и созданию сквозных кросс-организационных бизнес-процессов разработчики таких систем сталкиваются с проблемами, которые слабо изучены даже в области распределенного программирования. Так как система, поддерживающая выполнение длительного, сложного производственного процесса, пересекающего границы предприятий (а с учетом развития современных сетевых технологий, даже границы стран и континентов), состоит из подсистем не только разных разработчиков, но и разных областей ответственности и управления. Между такими подсистемами бывает невозможно создать единый центр координации, они могут требовать выполнения специфического ручного труда или необратимых действий, затруднены их тестирование и отладка, а ошибка работы любого участника может затронуть всех остальных. Ввиду этих и других причин цена

ошибки разработки системы может быть очень высокой. Тем не менее, в настоящий момент не существует единых стандартов и средств обеспечения надежности и качества подобных распределенных разнородных систем. Они по своей природе существенно сложнее стандартных программных средств, технических устройств и даже локальных автоматизированных информационных систем предприятия, так как могут включать в себя и эти перечисленные объекты, и другие, любые среди тех, что могут быть вовлечены в бизнес-процессы организационно и юридически независимых предприятий.

Простота и низкая стоимость разработки интеграционных решений позволяют создавать сложные распределенные системы дешево и с высокой скоростью, что приводит к росту объема выполняемых такими системами операций и увеличению их роли на предприятии. Обоснованно можно предположить, что планирование таких систем должно осуществляться особенно качественно и скрупулезно. Однако современные инструменты облегченной интеграции систем имеют и оборотную сторону своего использования: они сосредотачиваются только на технической реализации, не рассматривая системные вопросы взаимодействия независимых субъектов бизнес-процесса, не обеспечивая разработчиков средствами точного проектирования, не требуя большого объема специфических знаний и высокой квалификации программистов. Поэтому этап проектирования и расчетов предполагаемой надежности интегрированной системы ввиду описанных причин часто пропускается, что опасно при систематической разработке, использовании необратимых операций или вовлечении множества независимых участников.

Таким образом, актуальность данного исследования обусловлена необходимостью разработки методики анализа надежности, а также программного обеспечения, которое позволит проектировать, моделировать и рассчитывать надежность распределенных сервис-ориентированных систем, выполняющих различные процессы внутри и между взаимодействующими предприятиями.

Степень научной разработанности проблемы. Ввиду того, что проблемная область сформировалась как самостоятельное направление развития информационных систем и технологий в последнее десятилетие (период 2000-х – 2010-х), то остается много нерешенных проблем. К ним можно отнести исследование надежности выполнения кросс-организационных бизнес-процессов, обеспечение децентрализованного управления длительными транзакциями, анализ показателей работы слабосвязанных систем, разработка методик расчета надежности композиций разнородных компонентов информационных систем, сложность формулировки новой терминологии и т.д. Для решения перечисленных проблем можно воспользоваться результатами тех

областей знаний, где сходные проблемы имеют хорошую теоретическую проработку, математические и программные решения. Прежде всего, это решение проблем надежности технических систем, теория транзакций в системах управления базами данных и разработанная в 1980-х гг. теория процессного управления. Исследования взаимодействия компонентов, объединенных в бизнес-транзакцию, начали развиваться с 2000-х гг. в работах M. Little и M. P. Papazoglou. Кроме этих областей в ходе исследования были использованы результаты российских и зарубежных специалистов по теории длительных транзакций и экономических операций, надежности и моделированию бизнес-процессов и сервис-ориентированных систем, изложенных в работах: W.M.P. van der Aalst, M. Dumas, A.H.M. ter Hofstede, H. A. Reijers, M. Reichert, K. Jensen, M. Little, P. Massuthe, G. Alonso, M. La Rosa, J. Cardoso, F. Tang, H. Smith, P. Fingar и других.

Исследование затрагивает вопросы:

- построения и интеграции распределенных процессно-ориентированных систем, описания и моделирования бизнес-процессов;
- технологий сетевых сервисов, проблем построения и функционирования сервис-ориентированной архитектуры;
- теории расширенных транзакций, ее применения в области бизнес-процессов;
- надежности технических, организационных и программных систем в общем, а также вопросы надежности сетевых сервисов и их композиций;
- теории сетей Петри и ее применения к моделированию и анализу программных и информационных систем;
- теории однородных цепей Маркова и ее применения к анализу поведения и достижимости целевых состояний системы.
- разработки программного обеспечения для моделирования и оценки характеристик проектируемых бизнес-процессов.

Объектом исследования является распределенная информационная система, включающая разнородные компоненты, построенная с использованием сервис-ориентированных технологий и обеспечивающая выполнение бизнес-процессов предприятия(-й).

Предметом исследования является анализ надежности взаимодействия объектов распределенной сервис-ориентированной программной среды, которые совместно обеспечивают выполнение бизнес-процессы предприятия(-й).

Цель и задачи диссертационной работы. Целью работы является разработка методики и программного обеспечения для анализа надежности бизнес-транзакций в сервис-ориентированной среде. Это позволит создавать распределенные процессно-

ориентированные системы заданного уровня надежности и упростить процессы принятия решений в области обеспечения надежности проектируемых бизнес-транзакций в сервис-ориентированной среде.

Достижение указанной цели связано с решением комплекса следующих задач:

- Систематизировать информацию о современном состоянии технологий, методов построения и анализа надежности распределенных процессно-ориентированных систем.
- Описать неделимую последовательность операций распределенного взаимодействия как самостоятельный объект, определить его свойства, типы и принципы построения.
- Выявить показатели надежности распределенного взаимодействия и разработать методику ее анализа с точки зрения системного подхода.
- Разработать инструмент моделирования взаимодействия объектов распределенной среды, который учитывает его сложную природу и поддерживает описание сервис-ориентированных технологий построения систем.
- Разработать методическое и программное обеспечение, позволяющее осуществлять анализ надежности распределенного взаимодействия, и провести его апробацию на примере проектируемого взаимодействия разнородных систем.

Методы исследования. В диссертационной работе использовались методы системного анализа, методы теории множеств, окрашенных сетей Петри, цепей Маркова и теории надежности, методы математического и имитационного моделирования, объектно-ориентированного, сервис-ориентированного моделирования систем.

Научную новизну составляют следующие результаты, выносимые на защиту:

- Предложена система показателей надежности бизнес-транзакции, выявленных путем описания ее тройственной природы: как неделимого действия, композиции взаимодействующих систем и бизнес-процесса.
- Разработан инструмент моделирования бизнес-транзакций на базе окрашенных сетей Петри, учитывающий их сложную природу и современные подходы к построению сервис-ориентированных систем.
- Предложен алгоритм анализа устойчивости бизнес-транзакции на основе цепей Маркова.
- Разработана методика и программное обеспечение анализа надежности бизнес-транзакции на основе имитационного моделирования и предложенных показателей надежности.

Практическая и теоретическая значимость работы заключается в том, что:

- Формализовано понятие бизнес-транзакции: в целях анализа надежности распределенной сервис-ориентированной системы предложено рассматривать неделимую последовательность операций такой системы или неделимый акт ряда взаимодействий ее подсистем как самостоятельный объект, отражающий цели существования системы и внутренние характеристики ее функционирования.
- Предложено рассматривать бизнес-транзакцию как сложную разнородную систему, имеющую тройственную природу, что позволяет интегрировать подходы к оценке надежности из разных прикладных областей.
- Разработан инструмент на базе окрашенных сетей Петри, который позволяет описывать распределенные сервис-ориентированные системы, механизмы композиции сервисов, их отказов, операций восстановления и компенсации. Он может использоваться в процессах сервис-ориентированного программирования и проектирования сервис-ориентированных информационных систем.
- Предложены показатели надежности бизнес-транзакции, которые учитывают ее сложную природу и поведение. Показатели могут быть рассчитаны с помощью разработанной методики и программного комплекса имитационного моделирования.
- Предложен алгоритм обнаружения тупиковых ситуаций и бесконечных циклов на основе представления модели бизнес-транзакции в виде цепи Маркова.
- Создано алгоритмическое обеспечение и программный комплекс, которые позволяют проводить анализ надежности распределенных систем любой сложности, выявлять «узкие места» и предлагать способы повышения уровня надежности.
- Результаты работы используются на предприятии «ООО Группа компаний Инвест38» в задачах проектирования распределенных программных систем, моделирования сложных бизнес-процессов и оценки надежности взаимодействия со сторонними организациями.
- Результаты работы используются в учебном процессе по дисциплинам «Разработка распределенных систем», «Распределенные системы обработки информации», «Интернет-программирование» по специальностям «Прикладная информатика (в экономике)» и «Бизнес-информатика» ФГБОУ ВПО «БГУЭП».

Апробация работы

Теоретические и практические результаты были представлены на международном симпозиуме «Надежность и качество» в г. Пенза, 2013; на II международной научно-практической конференции «Инновационные информационные технологии» в г. Прага (Чехия), 2013; на международной научно-практической конференции «Информационные технологии и информационная безопасность в науке, технике и образовании «Инфотех-2013» в г. Севастополь, 2013; на английском языке на III международной практической конференции «Innovative Information Technologies» в г. Прага (Чехия), 2014; на 10-й международной научно-практической конференции «Инновации на основе информационных и коммуникационных технологий» в г. Сочи, 2013; на 11-й международной научно-практической конференции «Инновации на основе информационных и коммуникационных технологий» в г. Сочи, 2014; а также на ежегодных научных конференциях аспирантов кафедры информатики и кибернетики ФГБОУ ВПО «БГУЭП» с 2009 по 2013 г. и научных семинарах преподавателей кафедры информатики и кибернетики ФГБОУ ВПО «БГУЭП» в 2010 и 2013 гг.

Публикации

По результатам исследований опубликовано 19 печатных работ, из них 9 в журналах, рекомендованных ВАК, получено 1 свидетельство Роспатента об официальной регистрации программы для ЭВМ на программный комплекс для анализа надежности, реализующий предлагаемую методику, разработано 1 учебное пособие.

Структура работы и объем работы

Диссертация состоит из введения, трех глав, заключения, списка литературы из 244 наименований и двух приложений. Общий объем работы составляет 216 страниц, из которых 182 страницы основного текста, включает 50 рисунков и 25 таблиц.

1. Бизнес-транзакции в сервис-ориентированных информационных системах

Так как одним из основных объектов диссертационного исследования является взаимодействие интегрированных систем, то прежде чем описывать детальные свойства этого взаимодействия, проблемы моделирования и проектирования, опишем среду ее функционирования – распределенная программная система, включающая разнородные компоненты, построенная с использованием сервис-ориентированных и выполняющая бизнес-процессы предприятия.

1.1. Распределенные информационные системы

1.1.1. Процессно-ориентированные информационные системы

С начала 90-х годов в теории менеджмента организации особую популярность приобрел процессный подход к управлению. Он ориентируется на построение модели организации, ориентированной на выполнение бизнес-процессов, и пришел на смену функциональному подходу, построенному на принципах узкой специализации и жесткой иерархической структуры. При функциональном подходе вся деятельность предприятия распределяется по ряду независимых подразделений. Процессный подход предполагает смещение акцентов от управления отдельными структурными подразделениями к управлению сквозными бизнес-процессами, охватывающими все предприятие и даже выходящими за его пределы. Поэтому в основе подхода лежит деятельность по выявлению, описанию и исполнению бизнес-процессов компании.

Под бизнес-процессом (процессом) понимают совокупность взаимосвязанных видов деятельности, которая по определенной технологии преобразует входы в выходы, представляющие ценность для клиента [1]. Под входом и выходом понимаются, соответственно, ресурсы, необходимые для осуществления процесса и его результат в виде продукта или услуги. Клиент является потребителем результата процесса. В соответствии с процессной моделью предприятия, каждый процесс может быть разделен (декомпозирован) на ряд вложенных бизнес-процессов, и такая декомпозиция может рекурсивно продолжаться до уровня элементарных бизнес-операций [2].

Моделью бизнес-процесса называется его формализованное описание, отражающее реально существующую или предполагаемую деятельность предприятия [3].

Появление новой идеологии управления неизбежно повлекло за собой разработку поддерживающих ее информационных технологий. Так, в [4] и в [5] вводится новое понятие: процессно-ориентированная информационная система (ПОИС) – это

информационная система, которая исполняет бизнес-процессы и/или управляет ими на основе процессной модели, привлекая трудовые, программные и информационные ресурсы. Типичным примером систем, реализующим основной функционал ПОИС, являются WFMS (Workflow Management Systems) – системы управления потоком работ.

1.1.2. Интеграция ПОИС

В настоящее время информационная система предприятия может насчитывать несколько разных приложений, выполняющих различные функции, работающие в рамках отдельных подразделений или сред. В [6], [7] отмечается, что подобная ситуация поддержки множества коммерческих, унаследованных или собственных приложений является типичной для крупных компаний.

При такой организации семантически схожая информация содержится в различных подсистемах в разных формах, что приводит к ее дублированию, избыточности и накапливаемой противоречивости. Если выполняемый процесс затрагивает несколько функциональных подразделений или отдельных приложений, то внесение изменений в одном приложении вызывает необходимость схожих операций в других программных средствах, т.е. перенос данных между программными системами происходит вручную. В этом случае снижается качество и скорость выполнения сквозных бизнес-процессов и возникают серьезные ограничения в реализации процессно-ориентированного подхода [8, 9]. Кроме данной, выделяют еще ряд причин, также обуславливающих необходимость внедрения интеграционных технологий [10], например, совместная работа в среде множества предприятий, где необходим информационный обмен с поставщиками, потребителями или государственными органами. В связи с указанными причинами проблема интеграции множества приложений в рамках одного или нескольких предприятий является актуальной, активно исследуемой и предполагает внедрение целого ряда организационных и технологических мер на предприятии.

Весь объем решений интеграционных проблем внутри предприятия связывают с термином EAI (Enterprise Application Integration), а совместная реализация межорганизационных процессов относится в компетенции B2B-интеграции [11]. Под EAI понимается набор методов и инструментов, которые позволяют разнородным приложениям взаимодействовать для осуществления общих задач. В [12] EAI позиционируется как средство развития монолитных ERP-систем в гибкие интегрированные решения за счет возможностей WFMS [13] и BPMS.

Таким образом, термин «интеграция», соединяя различные сферы деятельности и ресурсы предприятия, трактуется достаточно широко как «*объединение компьютерных систем, компаний, процессов и людей*» [6]. Программная интеграция определяется как разработка новых приложений и модификация существующих таким образом, что они могут предоставлять свои данные и функциональность другим приложениям [14]. *В рамках данной работы будем называть систему «интегрируемой», если она способна к интеграции с другими системами, и «интегрированной», если эта система является результатом процесса интеграции нескольких «интегрируемых» систем*. Введем также понятие «интероперабельности» (от англ. interoperability) как способности одной системы взаимодействовать с другими.

1.1.3. Технологии интеграции

Интеграция предполагает построение системы нового уровня функциональности: распределенной программной системы. В [15] было дано определение *распределенной программной системы как совокупности независимых программных систем, взаимодействующих с целью решения проблем, не решаемых каждой системой индивидуально*. Таким образом, любая интегрированная система может относиться к классу распределенных, так как за счет интеграции она получает функционал, не доступный ранее отдельным ее компонентам. Ввиду того, что данное свойство оказывает немаловажное влияние на дальнейшее исследование, рассмотрим основные принципы построения распределенных программных систем.

Существует несколько технологических подходов к решению задач интеграции и поддержки унаследованных систем. Все эти подходы являются последовательными эволюционными этапами по пути общего развития информационных технологий и устранения своих внутренних недостатков. Так, в [6], [12], [10], [16], [14] выделяется основное свойство интеграционных решений, развитие которого обуславливает по большей мере и развитие подходов к интеграции, – связывание. Понятие «связывания», как и многие определения, относящиеся к области архитектуры программных приложений (по оценке в [17]), не имеет единого определения. В [12] показано, что оно не является комплексом правил или спецификаций, поэтому, *под связыванием мы будем понимать стиль организации взаимодействия между интегрируемыми системами, который определяет степень взаимозависимости систем, обусловленной техническими и организационными аспекты интеграции* [18].

Выделяют два диаметрально противоположных вида связывания, и эволюция всех интеграционных технологий, на наш взгляд, последовательно перемещалась от одного к другому: сильное и слабое связывание[1] [6, 12, 8, 14, 16, 19].

Сильное связывание предполагает объединение приложений таким образом, что они находятся в прямой зависимости друг от друга: разделяют общие методы, интерфейсы или данные. Например, сильно связанными являются методы одного приложения: они выполняются в одном адресном пространстве, разделяют общие переменные, вызов и передача информации происходит с минимальным задержками [6].

В распределенной среде признаками сильного связывания обладают прямой сетевой вызов или удаленный вызов процедур (RPC), позволяющий разнородным процессам вызывать методы друг друга безотносительно к их расположению в сетевой среде. Таким образом, сильное связывание приводит к необходимости существования знаний одной интегрируемой системы о другой и наличию допущений между ними. Так, изменение функциональности одной системы приведет к нарушению условий взаимодействия и дестабилизации всей интегрированной системы целиком. Избежать этого можно только постоянно отслеживая все зависимости или зафиксировав развитие систем на достигнутом уровне.

Слабое связывание повышает независимость компонентов, и, тем самым, устойчивость к изменениям. Это происходит за счет введения ряда промежуточных интерфейсов и агентов, которые устраняют прямую зависимость и позволяют изменять реализацию отдельных программных частей.

В [14], [12] описаны следующие отличия между сильным и слабым связыванием (см. табл. 1).

Таблица 1 – Отличия слабого и сильного связывания

Фактор	**Слабое связывание**	**Сильное связывание**
Физическая связь	Непрямая, через промежуточное программное обеспечение	Прямая
Тип взаимодействия	Асинхронный	Синхронный
Передача сообщений	Документно-ориентированная	RPC
Путь передачи данных	Маршрутизируется	Жестко фиксирован
Объединение технологий	Гетерогенное	Гомогенное
Типы данных	Независимые	Зависимые
Определение синтаксиса	Публикуется в открытом доступе	По взаимному соглашению

[1] В литературе употребляется также термин «когезия» (cohesion) – «сцепление» в понимании «связывание».

Продолжение таблицы 1

Фактор	Слабое связывание	Сильное связывание
Связывание (компонентов во время выполнения)	Позднее, отложенное	Фиксированное и предварительное
Цель интеграции	Повторное использование, повышение эффективности	Широкая применимость

Таким образом, слабая связанность позволяет создавать более гибкие и масштабируемые системы, способные к постоянным изменениям, но влечет недостатки в виде сниженной скорости реакции и сложностей централизованного управления. С другой стороны, сильная связанность позволяется повысить уровень контролируемости системы, используя, например, существующие технологии транзакционного управления (например, используемые в распределенной среде протоколы т.н. двухфазной фиксации транзакции), что сейчас используется в монолитных ERP или CRM-системах [14], а в среде, когда бизнес-логика приложения распределяется по слабосвязанным компонентам системы (т.е. в среде интеграции бизнес-процессов), транзакции приобретают свойства, не позволяющие использовать стандартные техники [20].

Существуют возможности разработки программ, способность к интеграции которых закладывается еще при их разработке. К таким принципам разработки относится принцип повторного использования. Он является важной составляющей EAI как средства интеграции бизнес-процессов [8, 16, 10, 14]. Повторное использование позволяет сократить функциональную избыточность, снизить стоимость поддержки информационной инфраструктуры, избежать разработки лишних программных систем и провести эффективную интеграцию различных систем. Повторное использование обеспечивает слабое связывание, и, в отличие от рассмотренных ранее технологий интеграции, позволяет строить программные объекты, специально предназначенные для интеграции.

Современным и популярным подходом к построению повторно-используемых систем [21] является технология сетевых служб (или веб-службы, или веб-сервисы, англ. web-services). Эта технология разрабатывалась как замена компонентно-ориентированной разработки с учетом недостатков и ограничений последней [22, 23]. В основу данного подхода легло понятие службы, которая предоставляет некоторые функцию клиенту в ответ на его запрос. Исходя из того, что рассматриваемая технология была построена с ориентацией на интернет-среду, разработкой методологической основы в области веб-служб занимается консорциум W3C, известный исследованиями в области веб-ориентированных стандартов. По мнению разработчиков [24], *веб-служба – программная система, разработанная для поддержки взаи-*

модействия между вычислительными машинами через сеть, идентифицируемая с помощью URI, интерфейсы и связи которой описаны с помощью XML. Другие системы взаимодействуют с веб-службой посредством методов, описанных в интерфейсе через SOAP-сообщения, перемещаемые обычно по HTTP-протоколу с XML-сериализацией в связке с использованием других веб-ориентированных стандартов.

В [25] сравниваются компонентный и сервисный подход к интеграции приложений, на основании чего можно сделать вывод о том, что компонентная технология должна использоваться при интеграции приложений на момент их разработки, тогда как сервис-ориентированная может использоваться для функциональной (процессной) интеграции унаследованных и разнородных систем.

Проведенный анализ в [21] показывает, что технология веб-служб вместе с открытыми технологиями и стандартами является на данный момент наиболее перспективным средством, позволяющим достигать повторного использования, а за счет открытых стандартов и публикации интерфейсов реализуется слабое связывание.

Таким образом, любой процесс как последовательность операций можно представить в виде последовательности вызываемых веб-служб или «обернутых» в веб-службы функций различных систем внутри предприятия и за его границами, что позволяет реализовывать интеграцию бизнес-процессов и информационных систем предприятия. И, вследствие подобных свойств веб-служб, эта технология имманентно поддерживает возможность интеграции средств автоматизации бизнес-процессов, и, следовательно, рекурсивную интеграцию частных бизнес-процессов и бизнес-функций до уровня процессов предприятия.

1.1.4. Архитектура, ориентированная на службы

В [26], [27] показано, что развитие теории веб-служб способствовало появлению новой идеологии интеграции – сервис-ориентированной архитектуры (СОА).

В современной теории программной инженерии не существует точной терминологии касательно слова «архитектура программного обеспечения» [17, 28]. Буч в [29], практически впервые говоря о языке UML, дает по меньшей мере восемь определений архитектуры, причем все они относятся к архитектуре как абстрактному термину, не связанному с программным обеспечением.

Для целей диссертационного исследования воспользуемся определением из [17]: «архитектура – набор значимых решений по поводу организации программного обеспечения, набор структурных элементов и их интерфейсов, через которые компонуется система, вместе с их поведением, определяемым во взаимодействии между

этими элементами, компоновка этих структурных элементов и их поведений в постепенно укрупняющиеся подсистемы, а также стиль архитектуры, который направляет эту организацию – эти элементы и их интерфейсы, их взаимодействие и компоновку».

Терминологические проблемы остаются и при попытке определить значение СОА. В официальном стандарте [30] эталонной модели СОА, разрабатываемом консорциумом OASIS, говорится, что было выработано множество определений. Анализ специализированной литературы (например, [14], [31], [32], [33], [34], [26], [35], [36]) показывает, что определения СОА могут быть различны. В [33], [34], [26], [35], [36] показано, что понятие «архитектуры» в СОА может не означать исключительно программную архитектуру, а внедрение СОА затрагивает всю архитектуру предприятия, ориентируя ее на развитие сервисов.

Далее будем понимать *СОА как архитектуру распределенного приложения, в которой все функции определены как независимые сервисы с вызываемыми интерфейсами, и выполнение сервисов в определенной последовательности позволяет реализовать тот или иной бизнес-процесс.*

Развитие СОА как технологической идеологии выявило широкое понятие службы. Под службой понимается некоторая бизнес-задача, единица работы, которая производится для удовлетворения чьей-либо потребности [31, 14]. Все зависимости службы от среды четко определены с помощью интерфейса, который явно описывает ее возможности. Служба реализует определенный объем бизнес-функций, и последовательность выполнения таких служб позволяет реализовывать целые бизнес-процессы.

Службу можно представить в виде черного ящика. Входные параметры полностью определяют поведение текущего экземпляра службы. Службы обычно не сохраняют свое состояние, поэтому могут многократно использоваться для реализации определенной бизнес-функциональности. Такое повторное использование является ключевой особенностью СОА, которая означает и возможность работы со службой различных потребителей в разных контекстах, и возможность построения на базе этой службы новых повторно-используемых сервисов. Каждый процесс, составленный из последовательности служб, можно представить в виде службы более высокого уровня абстракции, также предоставляющей свой функционал для вышележащих процессов [6]. Такой процесс можно называть «композитной» службой [33].

Ввиду того, что службы могут реализовываться различными корпоративными приложениями, а бизнес-процесс может включать множество служб, взаимоинтеграция информационных систем становится не целью внедрения СОА, а последствиями

обеспечения сервис-ориентированного подхода к выполнению бизнес-процессов. Стандартные методы интеграции позволяют выполнять ее между ограниченным количеством приложений и в определенном направлении быстро и недорого, однако при возрастании количества необходимых путей интеграции, изменении направлений движения данных или вовлекаемых процессов, стоимость интеграционных решений увеличивается пропорционально объему работ. СОА позволяет изначально сформировать из информационных систем ряд «строительных» блоков, которые можно связывать друг с другом на базе открытых стандартов и общепринятых технологий. Кроме того, основные поставщики корпоративных информационных систем и прикладных решений предоставляют возможности сервисной ориентации своих средств, за счет, например, поддержки технологий веб-служб (например, платформа 1С, начиная с версии 8.1).

На основе изложенного можно сделать важный вывод, используемый в дальнейшем исследовании. Сервисная ориентация – это парадигма проектирования информационных технологий и процессов, направленная на разработку самостоятельных слабосвязанных единиц работы (бизнес-служб), которые могут быть совместно и повторно использоваться для реализации целей бизнеса [37]. Представление всех функций приложений как самостоятельных сервисов с последующим их объединением в различные процессы позволяет создавать гибкие программные решения, легко адаптирующиеся к изменениям [31, 38].

1.1.5. Свойства сервис-ориентированной архитектуры

Опишем основные характеристики технологические характеристики сервиса в СОА [15, 39, 10, 33, 40, 37, 26, 41, 42]:

- Функции программной системы определены как службы. Сервисом может быть как отдельное приложение, так и его модули, процедуры или функции. Приложение может состоять из нескольких сервисов и само, в свою очередь, является службой.
- Службы можно комбинировать и перестраивать в новые бизнес-процессы, различные решения и сценарии в соответствии с новыми требованиями бизнеса.
- Слабая связанность. Службы выполняют определенную работу по запросам от других служб и возвращают определенный результат. Детали их реализации полностью скрыты, поэтому, как и в случае с компонентами, внутреннюю структуру и реализацию службы можно свободно изменять.

- Модульность служб. При переходе к СОА бизнес-функции и операции низкого уровня представляются в виде служб, которые могут компоноваться во все более высокоуровневые сервисы, которые, в конечном итоге, реализуют тот или иной бизнес-процесс.
- Сервис связан со средой через интерфейс. Как в случае с компонентно-ориентированным программированием, интерфейс описывает все зависимости службы от внешней среды. Он описывает все операции, которые может выполнять служба, все операции, выполнение которых она может затребовать в процессе своей работы, все входные и выходные данные.
- У служб нет состояний. Часто (но не всегда) службы не зависят от окружающей среды и не меняют своего поведения в зависимости от состояния других служб. Каждый очередной процесс коммуникации со службой не зависит от предыдущих. Однако сервисы могут изменять состояние других объектов, например, производить запись в базе данных или передавать информацию другим системам.
- Повторное использование служб. После разработки любой из сервисов можно использовать повторно для решения новых задач. Например, служба, отвечающая за печать определенных документов, может использоваться разными бизнес-процессами.
- Облегченная интеграция. Благодаря тому, что СОА строится с учетом открытых стандартов и технологий, нет необходимости в дополнительном программировании средств интеграции. Каждая служба в СОА поддерживает общеизвестные протоколы взаимодействия.
- СОА не привязана к определенной технологии, операционной среде или языку программирования. СОА может быть реализована с использованием различных технологий программирования, интеграции и распределенного взаимодействия.
- Не имеет значение расположение служб. Благодаря ориентации на открытые стандарты, распределенные реестры и интерфейсы, службы в СОА могут быть территориально распределены и, возможно даже, не все подключены к работе в определенный момент времени.
- Динамическое обнаружение служб. За счет использования реестров служб и открытого описания интерфейсов, служба может быть обнаружена, подключена и использована приложением во сразу время выполнения.

Исходя из того, что обращение к службам в той или иной последовательности позволяет, в конечном итоге, реализовать бизнес-процесс, а, следовательно, цель

СОА – обеспечение выполнения бизнес-процессов компании, то в контексте СОА часто ведется речь о BPM (например, в [43]), как методологии, позволяющей выявить бизнес-процессы, а, значит, определить необходимый контекст для реализации служб: в зависимости от точки зрения любой бизнес-процесс в СОА можно представить сервисом, а сервис – бизнес-процессом.

Интегрированная ПОИС на базе СОА сможет объединить все процессы организации, поддерживать их развитие и изменение, а также обеспечивать B2B-взаимодействие со сторонними предприятиями. В такой системе использование BPM как процессной методологии позволяет обеспечить управление процессами предприятия [44, 45, 46, 47, 48, 49], поддержку их развития, BPMS – их проектирование и разработку, а СОА, предоставляя техническую основу, позволит реализовывать разработанные процессы «на ходу» [21].

Таким образом, понятие сервис-ориентированной ПОИС позволяет описывать бизнес-процессы в терминах независимых служб, и, за счет ориентации на повторно-используемые сервисы, исполнять эти процессы в рамках отдельных программных систем и объединять их результаты в виде обобщенного процесса. Следовательно, парадигма СОА дает возможность внедрять концепции ПОИС «поверх» существующих разрозненных информационных систем, объединяя их в единую, интегрированную среду исполнения бизнес-процессов предприятия.

1.1.6. Методы моделирования сервис-ориентированной архитектуры и бизнес-процессов

По различным оценкам (например, в [33], [28], [27], [34], [50], [26]), внедрение как СОА, так и отдельных технологических решений, связанных с этой архитектурой, в том числе и разработка бизнес-транзакций, влечет целый ряд проблем и рисков, частично нивелировать которые можно еще на стадии проектирования и планирования будущей системы. Решение этих задач не только для сервис-ориентированных, но и вообще информационных систем неотрывно связано с аспектами их моделирования [51, 52, 53]. Так, в российской и зарубежной литературе выделяют подробную классификацию моделей и технологий моделирования (например, в [54], [55], [56], [57]), однако на практике для моделирования предметной области, структуры, содержания и поведения информационной системы используют достаточно небольшой перечень общеизвестных и общепринятых техник, таких как SADT (IDEF), DFD, ARIS, UML, BPMN, EPC [33, 34, 58, 59, 60, 62].

Все перечисленные подходы, являясь, по сути, простыми графическими нотациями, позволяют строить прикладные модели системы, удобные для непосредственно-

го восприятия человеком и командной работы, но не позволяющие проводить тщательные исследования модели, например, имитационное моделирование с использованием вычислительной техники. Так, в [63], например, отмечается существующее разделение всех методик моделирования в области информационных систем на две группы: группу «академических» языков моделирования, способных строить формальное описание системы, и группу «бизнес»-техник, которые были перечислены нами ранее.

Между тем, существует формализм, способный описывать поведение сложных систем и обладающий преимуществами как достаточно простого графического отображения, так и серьезной математической обоснованности, используемый как в практике проектирования информационных систем [33, 53, 64, 65], так и широко применяемый в разработке различных теоретических аспектов их функционирования – сети Петри. Несмотря на то, что эту методологию относят, в первую очередь, к «академическим» языкам, существует подходы к отображению различных «бизнес»-моделей к моделям на языке сетей Петри и наоборот (например, в [63], [66], [67], [68], [69]). Широкое развитие прикладных моделей и высокоуровневых сетей Петри целей разработки ПО и информационных систем позволяют предположить перспективность применения этой методологии к изучению сервис-ориентированных систем.

В [64], [68], [70], [71], [72], [69], [73], [74], [75] приводятся аргументы, позволяющие считать сети Петри подходящим средством моделирования информационных и производственных систем, систем класса workflow и сервис-ориентированных приложений:

- Возможность представления в модели в естественном виде параллельных процессов, их синхронизацию, причинно-следственную связь событий, проблемы распределения ресурсов, конкуренцию за них и т.д.
- Локализация состояния и действий, позволяющее моделировать системы как «сверху-вниз», так наоборот, то есть последовательно уточнять модель системы, разбивать ее на подсистемы, повторно использовать ранее рассмотренные компоненты и пр.
- Интепретируемость модели предполагает возможность ассоциировать модель с широким кругом различных значений и связей с моделируемыми объектами внешнего мира. Различные вариант интерпретации необходимы для решения различных целей модели: верификации, оценки производительности, планирования и пр.

- Графическое представление модели, облегчающее документирование и мониторинг системы, а также обмен информацией о системе между заинтересованными лицами.
- Формальная и точная семантика моделей позволяет проводить строгий анализ, автоматизированную реализацию или генерирование кода для контроля или имитации поведения системы. Как математический инструмент, модели на сетях Петри могут быть описаны набором линейных алгебраических равенств или другими математическими моделями, описывающими поведение системы. Это позволяет проводить формальную проверку свойств, связанных с поведением системы, например, параллельных операций, синхронизации, отсутствия тупиков, взаимных исключений и распределения ресурсов.
- Широкая поддержка научного и инженерного сообщества предполагает доступ к большому количеству техник анализа моделей, независимость от конкретного разработчика или поставщика, а также высокий уровень надежности разработанных методик и техник.

Анализ литературы показывает, что практически все сопутствующие понятия, раскрывающие парадигму СОА, подвергаются тщательному изучению, моделированию и развитию, в том числе и с помощью сетей Петри. В первую очередь, к этим понятиям относятся сервисы [76, 75, 77][1], веб-сервисы [57]), различные аспекты их взаимодействия [78, 79, 80, 81], технологии моделирования СОА [60], теория управления системами с множеством параллельных процессов [82, 83, 84, 85], теория управления бизнес-процессами и workflow [74, 66, 4, 86, 87, 88].[2]

1.2. Надежность сервис-ориентированной ПОИС

В связи с тем, что сервис-ориентированная ПОИС за счет интеграции объединяет функции нескольких информационных систем, приобретая черты самостоятельной системы, и реализует основные бизнес-процессы компании, то вопросы обеспечения качества ее работы занимают первостепенное значение. Государственный стандарт ГОСТ Р ИСО/МЭК 9126-93 «Информационная технология. Оценка программной продукции. Характеристики качества и руководства по их применению» [89], основанный на стандарте ISO/IEC 9126, определяет качество программной системы как «весь объем признаков и характеристик продукции или услуги, который

[1] Ввиду большого объема литературы, посвященной моделированию различных аспектов СОА, здесь приводятся только примеры исследований тесно связанных с сетями Петри.

[2] Отметим, что из данного обзора намеренно исключены аспекты исследований, относящиеся в первую очередь не только к СОА, например, различные аспекты работы с БД, теория передачи данных, теория многоагентных систем и пр.

относится к их способности удовлетворять установленным или предполагаемым потребностям».

В стандарте выделяются следующие характеристики качества:

- Функциональные возможности;
- Надежность;
- Практичность;
- Эффективность;
- Сопровождаемость;
- Мобильность.

По мнению Липаева в [90] количественным измерениям в наибольшей степени доступны только эффективность и надежность программного обеспечения. Из-за разнородного характера компонентов СОА, их независимости, асинхронных техник взаимодействия и широких возможностей интеграционных решений, привлекаемых в рамках этой архитектуры, надежность является одной из ключевых проблем, которая сопутствует внедрению и сопровождению СОА на предприятии.

1.2.1. Общие показатели надежности

Для оценки надежности организационно-технических систем в 60-80-х гг. прошлого века в СССР и за рубежом активно разрабатывалась т.н. теория надежности, прикладная техническая наука, занимающаяся изучением методов повышения эффективности и надежности различных объектов в процессе их эксплуатации [91].

Выделяют три направления теории надежности – математическая, изучающая математические закономерности отказов и методы количественного измерения надежности, статистическая, изучающая статистические характеристики, и физическая, исследующая физические причины, влияющие на надежность [92]. Исходя из этого, разумно предположить, что надежность преимущественно программных систем может изучаться только математической и статистической теорией надежности, тогда как физическая надежность вычислительных систем и сетей передачи данных играет второстепенную роль.

В этой теории надежность понимается как «свойство объекта сохранять во времени в установленных пределах значения всех параметров, характеризующих способность выполнять требуемые функции в заданных режимах и условиях применения, технического обслуживания, ремонтов, хранения и транспортирования» [93]. Аналогичные по смыслу определения предполагают стандарты «ГОСТ 13377-75. Надежность в технике. Термины и определения» (устар.) и «ГОСТ 27.002-89. Основ-

ные понятия. Термины и определения», но «ГОСТ Р 53480-2009. Надежность в технике. Термины и определения» [94], исключает, однако, требования «хранения и транспортирования», что позволяет использовать это определение надежности и для объектов информационной природы.

Кроме того, с понятием надежности объекта связывают отдельные его свойства [95, 96, 91], такие как:

- Безотказность – свойство объекта сохранять непрерывную работоспособность в течение некоторого времени.
- Долговечность – свойство объекта сохранять работоспособность до наступления предельного состояния с учетом необходимых перерывов на обслуживание и ремонт.
- Ремонтопригодность – приспособленность объекта к предупреждению, обнаружению и устранению отказов и неисправностей.
- Сохраняемость – способность объекта сохранять значение показателей работоспособности в течение и после хранения и (или) транспортирования. Срок сохраняемости – временной промежуток, в течение которого объект «сохраняет» свойство сохраняемости.
- Восстанавливаемость – свойство объекта, заключающееся в возможности проведении ремонтных работ после возникновения отказа с целью восстановления работоспособности. Время восстановления – продолжительность восстановления работоспособности объекта. В теории надежности рассматриваемые объекты в зависимости от этого свойства делят на восстанавливаемые и невосстанавливаемые.

Ключевым понятием теории надежности является отказ – событие, заключающееся в нарушении работоспособности объекта. Самоустраняющийся отказ называется сбоем. Наработка на отказ – продолжительность или объем работы объекта до его отказа.

В таблице 2 указаны некоторые статистические показатели надежности [97, 98, 94, 99].

Таблица 2 – Показатели надежности

Название	**Формула**	**Описание формулы**
Показатели безотказности		
Вероятность безотказной работы	$X = \frac{A}{B}$	A – число работоспособных изделий к концу испытаний; B – общее число изделий, поставленных на испытания.

Продолжение таблицы 2

Название	Формула	Описание формулы
Средняя наработка на отказ	$X = \frac{A}{B}$	A – суммарная наработка; B – суммарное количество отказов.
Интенсивность отказов	$X = \frac{A}{B}$	A – количество отказавших в процессе испытаний объектов; B – количество работоспособных объектов по результатам испытаний.
Частота отказов	$X = \frac{A}{B}$	A – количество отказавших объектов; B – суммарное количество поставленных к испытанию объектов.
Показатели долговечности		
Средний ресурс	$X = \frac{\sum_{i=1}^{B} A_i}{B}$	A_i – ресурс i-го объекта (наработка до отказа или общая наработка для восстанавливаемых объектов); B – суммарное количество объектов.
Средний срок службы	$X = \frac{\sum_{i=1}^{B} A_i}{B}$	A_i – срок службы i-го объекта; B – суммарное количество объектов.
Показатели ремонтопригодности		
Среднее время восстановления	$X = \frac{\sum_{i=1}^{m} A_i}{B}$	A_i – время восстановления i-го объекта; B – суммарное количество объектов; m – число отказов объекта.
Вероятность восстановления – вероятность того, что отказавшее изделие сможет быть восстановлено за время t	$X(t) = \frac{A}{B}$	A – количество объектов, восстановленных за время, меньшее t; B – количество объектов для восстановления.
Показатели сохраняемости		
Средний срок сохраняемости	$X = \frac{\sum_{i=1}^{B} A_i}{B}$	A_i – срок сохраняемости i-го объекта; B – суммарное количество объектов.

Для восстанавливаемых и невосстанавливаемых объектов номенклатура рассчитываемых показателей надежности изменяется соответственно природе объекта.

В [98] отмечается, что первоначально теория надежности рассматривала только технические системы, однако вследствие развития автоматизированных систем управления, важной задачей становится также исследование вопросов надежности систем взаимодействия «человек – техника», программных систем. Кроме того, на наш взгляд надежность любой информационной системы можно рассматривать как надежность составляющих ее видов обеспечения: технического, программного, математического, организационного, правового, информационного эргономического и т.д. Однако исследование надежности этих видов обеспечения выходит за пределы исследования не только теории надежности, но и, как в случае, например, с правовым обеспечением, области технических наук в целом. Отдельные авторы (например, в [100], [101]) рассматривают ограниченное количество видов обеспечения ИС, однако комплексная работа представляется нетривиальной задачей, требующей привлечения специалистов из разных областей наук.

1.2.2. Надежность программного обеспечения

Исходя из того, что в ПОИС управление сосредоточено преимущественно в рамках программных системам, этому типу надежности стоит уделить большее внимание. Ряд исследователей отмечают, что применение методов теории надежности по отношению к надежности программных систем обусловлено рядом ограничений, например [102, 103, 104, 98, 105, 103, 104]:

- Методы теории надежности применимы только к программным средствам, функционирующим в реальном времени и непосредственно взаимодействующим с внешней средой.
- Первостепенными факторами, определяющими надежность программного обеспечения, являются дефекты и ошибки их разработки, тогда как физическая надежность, защищенность от внешних воздействий или эффект старения имеют второстепенное значение.
- Редкое разрушение программ и редкая необходимость их полной физической замены приводит к принципиальному изменению понятия как сбоя и отказа – ключевых понятий теории надежности, так и времени и методов восстановления после сбоя – ключевых понятий теории восстановления, подраздела теории надежности.
- Ошибки в программном обеспечении могут приводить не к отказу всей вычислительной системы, а к искажению результатов ее работы. При этом, в от-

личие от аппаратного обеспечения, программное обеспечение в момент своего отказа или сбоя вероятно не изменится.

- Ошибки программного обеспечения могут быть исправлены и более уже не повторятся. Ошибки, выявленные в одной системе, могут быть исправлены в однотипных системах еще до того, как они приведут к отказу.
- Среди разработанных методов повышения надежности акценты смещаются в сторону введения различного рода временной, программной и информационной избыточности, стоимость которой, по сравнению с физической избыточностью, невелика. Тогда как классическая теория надежности оперирует, прежде всего, избыточностью физической. Однако при наличии дефектов в программном обеспечении его резервирование не приведет к повышению надежности, что не позволяет оперировать этим методом в полной мере.
- Непредсказуемость дефектов и ошибок, а также низкая вероятность их обнаружения для достаточно надежных программных средств не позволяет эффективно использовать традиционные методы расчета характеристик надежности систем.
- Для программных средств не применимы традиционные методы испытаний на надежность систем путем физического воздействия на них. Так, Липаев в [102] рекомендует заменять физическое воздействие «на методы форсированного воздействия информационных потоков внешней среды».

Ввиду того, что отказ программного обеспечения может не приводить к выходу ее из строя, дадим новые определения: *отказ программного обеспечения – событие, заключающееся в нарушении работоспособности программы при определенных условиях*. Например, отказ функционирования для определенного набора входных данных. Как правило, отказ не проявляется при других условиях работы. *Авария программного обеспечения – выход его из строя, или отказ, повлекший за собой невозможность дальнейшей работы программного обеспечения при любых других условиях, или переход на существенно более низкий уровень работоспособности*. Авария программного обеспечения в нашем понимании соответствует понятию «полный отказ» традиционной теории надежности.

Однако зачастую авторы специализированной литературы, посвященной надежности информационных систем, не проводят различий между их надежностью и надежностью сугубо технических систем, предлагая оценивать надежность традиционными показателями по свойствам безотказности, ремонтопригодности, долговечности, сохраняемости (например, в [106], [100], [107]), оценивая надежность ИС в целом, не вникая в детали архитектуры и функционирования.

Надежность программного обеспечения рассматривается в отдельных работах Липаева [102], Майерса [108], Шуракова [109], Тейтера [110] и др. Определение надежности имеет также различные трактовки. В [109], определяя надежность, автор ссылается на традиционный термин теории надежности, поддерживаемый справочниками и государственными стандартами по надежности технических устройств. В [108] надежность понимается как вероятность безотказной работы программы, что соответствует определению из математической теории надежности, где надежность есть вероятность сохранения работоспособности системы [111], но не согласуется с определениями из ГОСТа (что, кстати, и отмечают переводчики и редакторы в [111]). В [110] предполагается, что программа надежна, если она удовлетворительно выполняет свои функции. Это определение используется в некоторых устаревших стандартах ИСО, и, как замечается в [89], современный стандарт относит определение к «функциональной возможности» или «качества функционирования», тогда как надежность в [89] и в ГОСТ 28806-90 «Качество программных средств. Термины и определения» [112] представлена как «совокупность свойств, относящихся к способности программного обеспечения сохранять свой уровень качества функционирования при установленных условиях за установленный период времени». Это определение позволяет точно формулировать и оценивать надежность программных средств. Стандарты включают в определение следующие свойства:

- Стабильность (завершенность) – совокупность свойств программного обеспечения, характеризующая частоту отказов, обусловленных ошибками в программном обеспечении.
- Отказоустойчивость – совокупность свойств программного обеспечения, характеризующая его способность поддерживать определенный уровень качества функционирования в случаях программных ошибок или нарушения определенного интерфейса.
- Восстанавливаемость – совокупность свойств программного обеспечения, характеризующая его возможность восстанавливать уровень качества функционирования и восстанавливать данные, непосредственно поврежденные в случае отказа, а также время и усилия, необходимые для этого.

В [106] под отказом программного обеспечения понимается недопустимое отклонение характеристик его функционирования от предъявляемых требований, что, вообще говоря, согласуется с понятием отказа из традиционной теории надежности, однако автор этим затрагивает область управления требованиями – неотъемлемую и важную часть программное инженерии. Оценка надежности программных средств описана в стандарте «ГОСТ 28195-89. Оценка качества программных средств. Общие

положения», однако здесь надежность определяется только двумя критериями: устойчивость функционирования и работоспособность, тогда как показатели, позволяющие оценивать отказоустойчивость, стабильность и восстанавливаемость «разбросаны» по 23 оценочным элементам из 5 групп метрик, которые входят в эти два критерия. Но только 4 оценочных элемента являются расчетными на основе статистических данных, тогда как остальные критерии оцениваются экспертно по качественной шкале «0,1». Расчетные и экспертные критерии обладают одинаковыми весами при суммарном расчете показателя критерия, что позволяет предполагать, явное превалирование экспертных оценок над статистическими. Между тем, экспертная оценка возможна только в случае изучения исходного кода и архитектуры программного средства и, таким образом, невозможна для, например, программного обеспечения с закрытым исходным кодом или включающее элементы стороннего ПО.

Поэтому рассмотрим иные возможности изучения показателей надежности программного обеспечения. Из стандарта ISO 9126 [113], его внутренних, внешних метрик и метрик качества в использовании в таблице 3 выделим показатели, которые можно измерить для сложного программного продукта, состоящего из разнородных компонентов, внутреннее устройство которых в силу различных причин изучить затруднительно.

Таблица 3 – Показатели надежности программного обеспечения

Название	**Формула**	**Описание формулы**
Показатели стабильности		
Плотность скрытых ошибок – как много проблем остается в программе, которые могут в будущем спровоцировать отказы?	$X = \frac{\lvert A_1 - A_2 \rvert}{B}$	A_1 – количество предсказанных ошибок; A_2 – количество обнаруженных ошибок; B – размер программы.
Плотность отказов на один тестовый пример – как много ошибок было выявлено в процессе тестирования?	$X = \frac{A_1}{A_2}$	A_1 – количество обнаруженных отказов; A_2 – количество испытанных тестовых наборов.
Разрешимость отказов	$X = \frac{A_1}{A_2}$	A_1 – количество исправленных отказов; A_2 – количество обнаруженных отказов.
Плотность ошибок	$X = \frac{A}{B}$	A – количество обнаруженных ошибок; B – размер программы.
Устранимость ошибок – как много ошибок было исправлено?	$X = \frac{A_1}{A_2}$	A_1 – количество исправленных ошибок; A_2 – количество обнаруженных ошибок.

Продолжение таблицы 3

Название	Формула	Описание формулы
Устранимость скрытых ошибок	$X = \frac{A_1}{A_3}$	A_1 – количество исправленных ошибок; A_3 – количество предсказанных скрытых ошибок.
Среднее время наработки на отказ – как часто программа отказывает во время работы?	$X = \frac{T_1}{A}$	T_1 – время работы; A – количество отказов, зафиксированных в это время.
Степень покрытия тестами	$X = \frac{A}{B}$	A – количество проведенных тестов; B – количество тестов, необходимых для покрытия требований.
Степень стабильности по тестам	$X = \frac{A}{B}$	A – количество успешно пройденных тестов; B – количество тестов, необходимых для покрытия требований.
Показатели отказоустойчивости		
Вероятность избежания аварий	$X = 1 - \frac{A}{B}$	A – количество аварий; B – количество отказов.
Вероятность избежания отказов	$X = \frac{A}{B}$	A – количество предотвращенных критических и важных отказов во время проведения тестирования на отказ; B – количество тестов на отказ.
Вероятность избежания некорректных операций – как много реализовано функций с возможностью избежания неверных операций?	$X = \frac{A}{B}$	A – количество предотвращенных критических и важных отказов; B – количество тестов на некорректные операции.
Показатели восстанавливаемости		
Доступность – насколько доступна система для использования в определенных период времени?	$X = \frac{T_o}{(T_o + T_r)}$	T_o – время работы; T_r – время восстановления.
Доступность	$X = \frac{A_1}{A_2}$	A_1 – количество успешных попыток использования программы; A_2 – общее количество попыток использования программы.
Среднее время простоя – каково среднее время простоя в случае отказа?	$X = \frac{A}{B}$	A – суммарное время простоя; B – количество аварийных отказов.
Возможность перезапуска – как часто система может перезапускаться за заданное время?	$X = \frac{A}{B}$	A – количество перезапусков системы за заданное время; B – общее количество перезапусков системы.
Среднее время восстановления	$X = \frac{\sum_{i=1}^{B} A_i}{B}$	A – время восстановления отказавшей системы; B – количество случаев восстановления.

Продолжение таблицы 3

Название	Формула	Описание формулы
Восстанавливаемость – насколько система способна восстановиться после сбоя?	$X = \frac{A}{B}$	A – количество успешных восстановлений по результатам тестирования; B – общее количество тестов на восстановление.
Эффективность восстановления	$X = \frac{A}{B}$	A – количество успешных восстановлений за заданное время; B – общее количество тестов на заданное время восстановления.

Из [114] можно выделить следующие показатели оценки:

- Стабильность: наработка на отказ при отсутствии рестарта, измеряемая в часах; степень покрытия тестами.
- Устойчивость: наработка на отказ при наличии автоматического рестарта; ресурсы на обеспечение надежности и рестарта, по отношению к общему количество задействованных в системе ресурсов.
- Восстанавливаемость: длительность восстановления, измеряемая в минутах.

Кроме того, Липаев предлагает 4-ую характеристику надежности, называемую «доступностью» (или пригодность, готовность) и измеряемую относительным временем работоспособного функционирования. В целом те метрики, отличные от метрик ISO 9126, не позволяют сравнивать программные продукты безотносительно к их масштабу, архитектуре и объему. Например, длительность восстановления может иметь разную критичность для разных продуктов.

Таким образом, сравнивая набор показателей надежности программного обеспечения и технических устройств, свойство стабильности программного средства и по показателям и по определению соответствует свойству безотказности технического устройства, показатели ремонтопригодности и комплексные показатели надежности относятся к свойству восстанавливаемости программного обеспечения, однако показатели сохраняемости и долговечности для программного обеспечения найдены быть не могут, так как программные средства не имеют материальной природы своего существования[1] и функционирования – надежность их носителей и условий функционирования уже не относится к надежности ПО.

Ввиду того, что предметом нашего исследования является надежность бизнес-транзакции как единичного акта взаимодействия, как некоторого идеального объекта, то для целей исследования не важны отдельные аспекты ненадежности среды ее функционирования, например, технические отказы, уязвимости программного обес-

[1] Безусловно, программное обеспечение физически расположено на определенном носителе, но свойства сохранности и долговечности присуще носителю информации, а не самой информации.

печения или случаи саботажа работы со стороны операторов. Предполагается, что каждый участник бизнес-транзакции может быть ненадежным, но не рассматриваются ни причины этой ненадежности, ни способы ее устранения. Изучение собственной надежности конкретных программных средств, устройств, персонала и другим видов обеспечения относится к соответствующим областям наук и не затрагивается данным исследованием. Таким образом, предполагается, что в потенциально ненадежной среде (безотносительно к причинам ненадежности) бизнес-транзакция должна реализовывать взаимодействие, надежность которого необходимо оценивать. Поэтому в работе предполагается, что повышение надежности взаимодействия можно достичь, изменяя структуру взаимодействия при неизменной собственной надежности его участников.

Поэтому отметим, что, на наш взгляд, комплекс показателей стандарта ISO 9126:2001 является наиболее перспективным для статистической оценки надежности сложных программных систем. В стандарте для свойства стабильности нет явного показателя «частота отказов», однако если рассмотреть подобный показатель из статистической теории надежности и представить каждый объект как «тестовый набор», то можно заметить «плотность отказов» по смыслу соответствует показателю «частота отказов». Коэффициент готовности соответствует доступности. Такие параллели можно обнаружить по другим показателям.

Кроме показателей надежности, рассчитываемых на основе полученных статистических данных, существует несколько моделей надежности программного обеспечения, позволяющих прогнозировать уровень стабильности программного обеспечения, таких как [110, 104, 106, 107, 115]: модель Шумана, Ла Падулы, Джелинского-Моранды, Шика-Волвертона, Муса, Миллса, Липова, Коркорэна, Нельсона, простая интуитивная модель, экспоненциальная модель и пр. Однако ряд динамических моделей использует ряд допущений и ограничений, сложных для реализации в крупных программных системах, и, как отмечается в [104], прогнозирующие способности моделей должны быть исследованы дополнительно. Статистические модели позволяют оценивать отдельные свойства надежности и программного обеспечения – предполагаемое оставшееся количество ошибок (модель Миллса, модель Липова) и вероятность безотказной работы (модель Коркорэна, Нельсона.). Так, в [110] показано соответствие модели Нельсона способам расчета вероятности безотказной работы, приведенной нами ранее, при условии нахождения вероятностей появления тех или иных наборов входных переменных в программу. В целом следует отметить, что выбор модели надежности ПО представляет собой сложную задачу, при решении которой необходимо принимать во внимание цели и задачи расчета надежности, имею-

щиеся ресурсы, возможности сбора статистических данных, квалификацию и опыт экспертов. Возможно, оценку надежности следует проводить несколькими методами одновременно.

Процессно-ориентированные системы разрабатываются с учетом возможности работы оператора – выполнении им «ручной» части бизнес-процесса. Для рассмотрения надежности работы оператора в традиционной теории надежности вводится понятие системы «человек-техника» – это частично автоматизированная техническая система, часть функций которой выполняется человеком.

Разделяют несколько подходов к расчету надежности подобной системы [105]: для систем с некомпенсируемыми ошибками оператора и отказами техники, для систем с компенсацией ошибок оператора и отказов техники, для систем с компенсацией ошибок оператора и последствий отказов техники. Исследуются проблемы дублирования операторов, факторов его безошибочной работы, утомляемости и скорости реакции. Влияние психофизиологических факторов на надежность оператора исследуются, например, в [116], [117], [118]. Для человека-оператора выделяют отдельные фазы работоспособности [98]: начало работы с низким уровнем скорости и точности, устойчивой работоспособности и утомляемости.

В качестве статистических показателей надежности оператора можно выбрать вероятность безотказной работы и вероятность своевременного выполнения операции. В любом случае, оценка надежности оператора, как участника процессного взаимодействия, является сложной процедурой, ввиду особенностей человеческой психики и физиологии требующей дополнительного анализа для каждого конкретного факта участия человека.

1.2.3. Надежность сетевых служб

В отличие от других подходов к интеграции, где выход из строя любого из компонентов, вовлеченных во взаимодействие, приводит к отказу всей системы, сервис-ориентированная концепция позволяет обеспечить более высокий уровень отказоустойчивости и восстанавливаемости за счет имманентных свойств сервисов: слабой связанности, отсутствия состояния, повторного использования, динамического обнаружения и привязки, облегченной интеграции, открытым стандартам взаимодействия. За счет этих свойств в случае отказа любого из компонентов инфраструктура СОА способна обнаружить и подключить к процессу сервис-замену (или группу сервисов). Таким образом, эффект снижения надежности из-за снижения стабильности

среды может быть нивелирован повышением восстанавливаемости и отказоустойчивости процессов.

С этой точки зрения, исследование надежности может быть проведено с учетом методов теории восстановления и резервирования традиционной теории надежности. Теория восстановления занимается изучением процессов восстановления и функционирования восстанавливаемых объектов, а теория резервирования изучает способы повышения надежности за счет привлечения дополнительных средств и возможностей, избыточных с точки зрения нормального функционирования системы. Но для сервис-ориентированных систем трудно провести границы между областями использования этих подходов: восстановление отдельной функции системы может не означать восстановление отказавшего элемента, его функцию может выполнять другой элемент или целый набор элементов, и, кроме того, ввиду стохастической природы исполняемых бизнес-процессов, участия в них «ручных операций», широкой, возможно, номенклатуры сервисов и подсистем невозможно заранее предсказать отказы системы и способы ее восстановления. Поэтому приложение теории резервирования и восстановления к сервис-ориентированным информационным системам требует отдельного исследования.

Для обеспечения надежности консорциумом OASIS был разработан ряд стандартов, связанных с сервис-ориентированным программированием: WS-Security, WS-ReliableMessaging, WS-Reliability и другие. С точки зрения этих стандартов надежность представлена обеспечением надежности передачи сообщений: гарантированной доставки, отсутствием повторений и должным порядком передачи, т.е. стабильности и отказоустойчивости передачи сообщений между сервисами [119, 7].

Однако кроме проблем передачи сообщений между сервисами, которые, строго говоря, связаны с промежуточным программным обеспечением, ответственным за доставку сообщений, консорциумом OASIS не рассматриваются проблемы надежности, связанные с отказом работы одного из сервисов, реализующих бизнес-процесс вместе с другими службами. Общие форматы взаимодействия таких сервисов в рамках одного процесса рассмотрены в стандартах WS-BusinessActivity, WS-AtomicTransaction, WS-Coordination, WS-Transaction, WS-CAF, WS-Choreography и других, но эти стандарты описывают общий порядок и характер, не детализируя поведенческие аспекты сервисов в случае отказа какой-либо из сторон интеграции. Или, в частности, в случае необходимости отмены результатов частично выполненного процесса.

1.3. Особенности исследования надежности бизнес-транзакций

На практике существуют такие бизнес-процессы, которые могут обладать признаками атомарности, т.е. должны быть выполнены полностью или не выполнены вообще. Особенно это важно для деятельности, охватывающей несколько подразделений или взаимодействующих предприятий, или для таких процессов, как, например, «купля-продажа», где невозможно только частичное выполнение [120]. В [120], [121], [122], [123] показано, что такой процесс, исполняемый при поддержке средств автоматизации, принимает черты транзакции в теории баз данных: он состоит из нескольких операций, которые должны быть выполнены все вместе или не выполнены вообще. Подобный процесс был назван бизнес-транзакцией.

Исследование надежности взаимодействия в сервис-ориентированной системе осложнено тем, что в подобном взаимодействии принимают участие сервисы, которые через бизнес-процессы олицетворяют технические, программные или организационные ресурсы предприятия. Поэтому необходимо перейти к объекту, описывающему взаимодействие, который, невзирая на природу и характеристики своих составляющих, позволяет анализировать совокупной надежность их работы. На наш взгляд, объектом, подходящим на эту роль, является бизнес-транзакция.

1.3.1. Системные транзакции

Транзакция в теории баз данных (далее «системная транзакция») – это логическая единица работы базы данных (БД), которая должна быть полностью выполнена или полностью отменена [124]. Транзакция переводит БД из одного согласованного (устойчивого) состояния в другое, но допускается, что во время выполнения транзакции БД может находиться не в согласованном состоянии [125]. Согласованное состояние БД – это такое состояние, при котором выполняются все условия ограничений целостности. Поэтому, чтобы гарантировать согласованность БД, транзакция должна начинаться только в устойчивом состоянии БД.

Любая системная транзакция основана на наборе базовых принципов, называемых ACID (Atomicity, Consistency, Isolation, Durability) [126]:

- Атомарность (Atomicity) предполагает, что транзакция либо должна быть выполнена полностью и до конца, либо не выполнена вообще.
- Согласованность (Consistency) предполагает, что транзакция переводит БД из одного устойчивого состояния в другое.

- Изолированность (Isolation) обеспечивает незаметность незавершенных действий транзакции для других транзакций. Иными словами, пока транзакция не завершена, никакие результаты ее действия не будут доступны, т.е. обеспечивается изоляция транзакции от других.
- Долговечность (Durability) указывает на то, что, если транзакция была зафиксирована, ее результаты устойчивы и сохраняются в БД даже несмотря на возможные последующие сбои.

Системные транзакции, поддерживающие ACID-свойства, называются «плоскими» [127]. Они являются основными «строительными» блоками, воспроизводящими атомарные действия над базой данных.

В стандарте языка SQL [128] определены 3 выражения для манипуляции транзакцией: COMMIT, ROLLBACK и SAVEPOINT, означающие подтверждение (фиксацию), откат и сохранение контрольной точки транзакции. Оператор COMMIT означает успешное завершение транзакции и указывает на фиксацию всех внесенных изменений. ROLLBACK производит т.н. откат транзакции, то есть отмену всех внесенных ею изменении до начального устойчивого состояния БД. Оператор SAVEPOINT позволяет установить в процессе выполнения транзакции ряд контрольных точек, к которым возможно провести откат в случае неудачности последующих изменений.

Обеспечение изолированности транзакций при их параллельном выполнении осуществляется с помощью технологий сериализации транзакций [125], блокировок различного уровня и типов, технологий временных меток и пр. При распределенных вычислениях широко используется т.н. протокол двухфазной блокировки (2PC-протокол) состоящий из двух этапов [124, 125]. На первом этапе (фаза «роста») транзакция накладывает блокировку на все используемые ресурсы. На втором этапе (фаза «сжатия») транзакция снимает блокировки после успешного внесения изменений. Для решения различных ограничений и проблем существуют модификации этого протокола, усиливающие или ослабляющие его.

Отдельной проблемой является управление транзакциям в распределенной или интегрированной среде, когда она выполняется множеством участников. Здесь в зависимости от структуры и задач распределенной системы могут применяться различные алгоритмы распределенного блокирования и фиксации транзакций [129]. Универсальным же решением проблем атомарности и изолированности таких транзакций является использование протокола двухфазной блокировки.

1.3.2. Бизнес-транзакции в распределенных ПОИС

В англоязычной научной литературе бизнес-процессы, являющиеся многоэтапными транзакциями, с одной стороны, и продолжительными процессами с признаками атомарности, с другой, называются «long-lived transaction», «business activity», «business transaction», т.е. длительная транзакция, бизнес-действие или бизнес-транзакция соответственно. Характер и применимость этих определений трактуется контекстом работ, в которых они используются. В рамках исследования будем называть такой процесс «бизнес-транзакцией».

В русскоязычной литературе, несмотря на использование этого термина в переводных изданиях, не дается явного определения. Например, в [130], [131] под бизнес-транзакцией понимается либо некая неделимая операция между сторонами, которая требует исполнения нескольких обычных транзакций, либо воспроизводится классический термин экономики – «трансакция», как минимальной логически осмысленной операции, которая может быть или совершена полностью, или полностью отменена. Однако не рассматривается понятие транзакции с точки зрения процессного управления. В [17] сравниваются два вида транзакций и отмечается, что системные транзакции адресованы приложением к СУБД, в то время как бизнес-транзакции адресуются пользователем приложению.

По аналогии с определением системной транзакции и определением из [122] положим, что *бизнес-транзакция – это согласованное изменение состояния двух и более сторон взаимодействия (бизнес-процесса), где каждая сторона готова к такому изменению и знает, что его согласованно примут все стороны.* Из требования обоюдного изменения состояния следует требование атомарности, а, исходя из того, что изменение состояния по масштабу может быть любым – размер бизнес-транзакции не ограничивается.

Бизнес-транзакции выполняют функции, являющиеся критическими для бизнес-процесса, например, управление цепочкой поставок, и могут координироваться множеством независимых партнеров, среди которых могут быть крупные отделы или даже отдельные предприятия. Каждый участник транзакции обладает собственным состоянием, которое согласовано с другими участниками. Вместе участники транзакции поддерживают в некотором состоянии общее поле взаимоотношений, которое соответствует распределенным ограничениям, принятым между сторонами, и, таким образом, бизнес-транзакция является согласованным переходом этих отношений к новому состоянию [121]. Поэтому бизнес-транзакция схожа с обычной транзакцией как семантически, которая представляет собой набор операций, переводящих базу

данных из одного устойчивого состояния в другое, так и технически, требуя для своего исполнения определенной среды и протоколов, поддерживаемых всеми участниками. Но многие авторы (например, в [120], [123]) отмечают, что бизнес-транзакции, в отличие от распределенных транзакций СУБД, управляются не только техническими требованиями, такими как координация, согласованность данных, способность к отмене и восстановлению, но и экономическими ограничениями. Например, цель транзакции достигается только тогда, когда все участники пришли к согласованному решению о том, что она закончена, например, проведена окончательная оплата за полностью оказанные товары и услуги.

Так, в описание транзакции входят такие данные:

- стороны и исполнители транзакции;
- объекты, над которыми производится транзакция, в т.ч. формы и документы, которые обрабатывает транзакция;
- назначение транзакции (платежей, доставки и пр.);
- ограничения, которые регламентируют любые аспекты транзакции, в т.ч. временные ограничения (максимальное время, в течение которого транзакция может быть активна);
- ограничения, внешние по отношению к транзакции и ее участникам, которые выражают требования закона, правила торговли и термины контракта, публичные, принятые политики, законы и регулировки, которые применимы для участников транзакции;
- особенности связи и взаимодействия с другими транзакциями.

К технической группе требований относят различные системные аспекты, связанные с исполнением транзакции и взаимодействием ее с внешней средой. Эти требования затрагивают не только бизнес-транзакции, но среду их исполнения:

- Поддержка длительных, композитных взаимодействий;
- Определение исключительных ситуаций и их последствий, включая последовательности восстановления начального состояния;
- Поддержка компенсируемых и восстанавливаемых транзакций;
- Использование альтернативных транзакций для выполнения одних и тех же функций;
- Возможность согласования и взаимодействия с другими транзакциями;
- Возможность поддержки различных моделей безопасности транзакций: целостности, конфиденциальность, неотказуемости и пр.;
- Возможность осуществления аудита, мониторинга и ведения журнала исполнения транзакций.

Ввиду этих требований бизнес-транзакции могут реализовываться как классическими средствами интеграции информационных систем, так и технологиями, обеспечивающими коммуникацию разобщенных и слабосвязанных систем (например, с помощью технологий веб-служб: [120] [123], [132], [133], [134], [135]).

Область управления транзакциями в СУБД хорошо изучена и строго формализована, и поэтому с начала 90-х годов, с момента появления программных средств, автоматизирующих выполнение бизнес-процессов, предпринимались неоднократные попытки отобразить модели системных транзакций к бизнес-транзакциям (как, например, в [125], [136], [137]). Однако было замечено (например, в [33], [138], [132], [120], [123], [139], [140], [141], [142], [143], [131]), что не все бизнес-транзакции могут удовлетворять ACID-параметрам (или выполнение этих свойств для транзакций может быть вообще не желательно).

В общем случае мы выделяем следующие отличительные свойства бизнес-транзакций:

1. Бизнес-транзакции, в отличие от обычных транзакций, могут выполняться в течение продолжительного периода, от нескольких часов до нескольких недель. И время выполнения бизнес-транзакции непредсказуемо. Например, запрос информации о наличии авиабилетов, и оплата полета может занимать пять минут, работа с онлайн-магазином – час, а сложные бизнес-транзакции, такие как заключение контрактов, могут длиться днями. Это свойство бизнес-транзакций позволяет определить такую транзакцию в качестве «длительной» или «длинной» (англ: long-lived transaction), которая, как отмечается в [129], может продолжаться несколько часов или дней. При этом бизнес-транзакция может быть подвергнута разделению на несколько этапов (как показывают, например, в [121], [144], [132], [139]), функциональность которых можно инкапсулировать в виде подтранзакций, и представить основную транзакцию композитной по своей структуре.
2. Одна из основных задач системы управления транзакциями – поддержка их сериализуемости, т.е. последовательного выполнения по заранее определенному плану, что гарантирует выполнение свойства изолированности. Бизнес-транзакции по природе своей могут выполняться как параллельно по отношению друг к другу, так и обладать внутренним параллелизмом на различных стадиях, что может привести к конфликтам изолированности и согласованности системы.

3. Бизнес-транзакции распределены в разнородной среде. В отличие от обычных транзакций, которые выполняются в рамках единой инфраструктуры поддержки и баз данных, бизнес-транзакции могут пересекать границы предприятий и исполняться в рамках информационных и технических сред с различными свойствами, ограничениями и условиями выполнения. В такой среде невозможно или затруднительно проследить за исполнением определенных ограничений, накладываемых ACID-свойствами.
4. Ввиду слабой связанности участников бизнес-транзакции невозможно предсказать их количество и качество работы. Параллельные экземпляры одной и той же транзакции могут задействовать различные группы и составы групп исполнителей для одинаковых операций.
5. Некоторые ресурсы, вовлекаемые в бизнес-транзакцию, обладают высокой динамикой. Например, результаты вычислений или коммуникационные ресурсы могут многократно изменяться в течение выполнения даже одной транзакции.
6. Бизнес-транзакция, будучи отображением бизнес-процесса с формальной точки зрения, предполагает возможность декомпозиции на ряд других бизнес-процессов (транзакций), каждый из которых также может быть декомпозирован. Например, идеологии BPM и COA предполагают, что одни и те же процессы на предприятии могут иметь разную реализацию разными исполнителями. Конкретный выбор исполнителя и его реализации процесса зависит от системы управления бизнес-процессами, настройки интеграционной, промежуточной среды и др. факторов. Таким образом, выполнение одной и той же бизнес-транзакции может приводить к инициации различных исполнителей и процессов.
7. Частично или полностью бизнес-транзакция может состоять из ручных операций или привлекать для своего выполнения персонал предприятия, который находится вне сферы контроля транзакции. Человеческое взаимодействие происходит асинхронно, с временными задержками и непредсказуемой производительностью.
8. Результаты многих операций (например, ручных) бизнес-транзакций невозможно автоматически отменить. Для возвращения системы к начальному состоянию могут использоваться специальные компенсирующие действия.

По этим причинам протоколы блокировки (в т.ч. протокол двухфазной фиксации), гарантирующие атомарность, не могут напрямую использоваться. Например, в [33], [34] отмечается, почему невозможно использовать традиционные подходы к реализации таких транзакций при сервис-ориентированной реализации:

- 2PC-протокол противоречит слабосвязанной природе служб: во-первых, не все вовлеченные во взаимодействие службы могут поддерживать транзакционное поведение (например, откат транзакции), во-вторых, для некоторых типов бизнес-транзакций в B2B-среде невозможно использование единого координатора, в-третьих, длительная природа транзакций не позволяет производить монопольную блокировку ресурсов.
- Принудительный и немедленный откат транзакции невозможен в силу ее экономической сущности: для этого необходимо явное одобрение всех участвующих сторон, которое зачастую невозможно получить в короткие сроки.
- Часть процессов бизнес-транзакции может быть осуществлена вручную.
- Откат бизнес-транзакции, реализуемой «крупнозернистым» сервисом (т.е. выполняемом на достаточно высоком уровне бизнес-процессов предприятия), который привлекает для своего выполнения множество других служб (процессов), требует также вовлечения всех участвующих сторон и сервисов.

Поэтому блокировка может заменяться компенсацией [34, 33, 145]. В этом случае предполагается, что все обновления транзакции – успешные и их фиксация происходит немедленно, однако заранее подготавливаются способы отмены изменений с помощью специальных компенсирующих действий. Каждый вложенный элемент бизнес-транзакции может обладать своей компенсирующей операцией, и отмена транзакции в таком случае будет происходить с помощью последовательного запуска всех компенсирующих операций в порядке, обратно выполнению вложенных элементов.

Компенсация применяется в качестве «семантического» отката транзакции, когда состояние системы приводится к начальному не за счет отмены изменений, произведенных транзакцией, а за счет иных действий, выполнение которых позволяет достичь желаемого состояния. Проводя параллель с бизнес-процессами можно полагать, что компенсация бизнес-процесса является другим бизнес-процессом, приводящим распределенную ПОИС к некоторому начальному состоянию.

1.3.3. Системные аспекты бизнес-транзакции

Несмотря на то, что ранее мы рассматривали бизнес-транзакцию как последовательный процесс изменения состояния ее участников, который в равной степени схож и с системными транзакциями, и распределенными бизнес-процессами, возникает необходимость представить бизнес-транзакцию как систему, что позволит рассмотреть ее структуру, связи, выделить отдельные элементы, свойства и ее поведение, а затем исследовать ее с позиций системного подхода.

Как показывается в [146] термин «система» используется, когда необходимо охарактеризовать исследуемый или проектируемый объект как нечто единое целое, сложное, о котором невозможно сразу дать представление, показав его, изобразив графически или описав с помощью математического выражения. В полной мере это утверждение, на наш взгляд, соответствует бизнес-транзакции: несмотря на простоту понимания ее общих принципов, бизнес-транзакция представляет собой сложный объект во множестве своих участников и их поведения, связей между ними, структуры взаимодействия, внешних и внутренних факторов. Исследователи в области теории систем показывают, что не существует сколько-нибудь однозначного, емкого и корректного определения системы. Существует большое многообразие определений, описанных, например, в [146], [147], [148], которое объясняется характером системных исследований и задачами, в рамках которых вводится понятие системы.

Воспользуемся определениями системы из [148] по Гибсону и из [147]: *бизнес-транзакция – интегрированная совокупность взаимодействующих элементов, являющаяся определенным целостным единством и предназначенная для кооперативного выполнения некоторой функции.* Не лишним будет показать, что бизнес-транзакция тогда соответствует определению [146], где под системой понимается конечное множество функциональных элементов и отношений между ними, выделенное из среды в соответствии с определенной целью в рамках определенного временного интервала.

В [147] отмечается, что сущность системы может быть раскрыта через выработанные практикой неформальные признаки, характеристики и классификации. Рассмотрим их применительно к бизнес-транзакциям.

Элемент системы – ее простейшая, неделимая часть [146]. В качестве элемента системы может выделяться участник бизнес-транзакции, которому принадлежит множество функций и сервисов, и отдельные функции и сервисы, объединяемые в некое единое целое для выполнения общего процесса, и частные процессы, выполняемые участниками. При этом не исключается, что каждый элемент бизнес-

транзакции, чем бы он не был представлен, вне рассматриваемой бизнес-транзакции может участвовать в других взаимодействиях, входя в состав сторонней системы.

Понятие подсистемы используется для многоуровневого расчленения системы. В бизнес-транзакциях, в зависимости от ее структуры, подсистемами могут быть, например, участники с областью их контроля или отдельные сервисы с множеством функций. Подсистемы бизнес-транзакции далее в зависимости от контекста будут называться элементами, подпроцессами или операциями бизнес-транзакции.

Пользуясь определениями из [146] и [149] скажем, что структура бизнес-транзакции представляется совокупностью устойчивых связей и отношений между ее элементами, описывает устройство (строение) бизнес-транзакции. В зависимости от характера элементов, отношений между ними, целей наблюдателя и системы структура транзакции может иметь различное воплощение. Например, если элементами транзакции выделяются функции сервисов, то они связаны отношениями передачи данных, которые в свою очередь ограничены интерфейсными контрактами. Не исключается возможности введения иерархии бизнес-транзакции, включающей другие подсистемы с собственными структурами. Возможно, оркестровка и хореография [150] являются типовыми структурами, единственно-возможными «граничными» формами организации бизнес-транзакции, тогда как все реальные структуры находятся во множестве всевозможных комбинаций элементов в пределах этих границ.

Понятие цели бизнес-транзакции отражается в приведенных нами определениях, где цель раскрывается через выполнение некоторой общей функции (цель существования распределенной системы) или как достижимое конечное состояние участников этой транзакции, результаты ее работы.

Предложенные ранее свойства бизнес-транзакции позволяют описать ее системные признаки. Воспользуемся признаками, определенными в [149] и [147]: Целостность – бизнес-транзакция представляет собой целостную совокупность элементов, различной природы и целей, но действующих согласно общей задачи, например, цели бизнес-процесса. Эта цель не может быть достигнута порознь, тем самым появляется новое качество, присущее только транзакции целиком (свойство эмерджентности) – возможность достижения этой цели. Кроме того, каждый элемент, играя свою роль в оркестровке или хореографии, получает новые возможности функционирования, зависящие от своих связей. Элемент вне рассматриваемой бизнес-транзакции изменит или потеряет эту часть своих качеств. Поэтому связанность бизнес-транзакции определяется исходя из ее внутренних связей и взаимодействий. Расчленимость предполагает возможность выделения элементов, описание их позиций и отношений на различных уровнях иерархии. С другой стороны, интегративные (не-

аддитивные) качества бизнес-транзакции предполагают, что никакие ее элементы и подсистемы не обладают некоторыми свойствами, которыми обладает эта транзакция в целостном виде.

Бизнес-транзакция в каждый момент своего существования обладает состоянием, которое определяется совокупностью свойств каждого элемента и участника (при условии, что передача информации между элементами БТ осуществляется мгновенно), это некоторая «мгновенная» фотография системы. Переход между состояниями, от начального до конечного, является признаком наличия поведения системы. Бизнес-транзакция функционирует, когда участники передают друг другу информацию, другие объекты, осуществляют операции над ресурсами и формируют некоторый значимый результат. Все это приводит их к определенному конечному состоянию, которое согласовано между всеми участниками. Такое свойство дает возможность ограниченно применять теорию системных транзакций для бизнес-транзакций.

Наличие перечисленных свойств показывается методами формализованного представления систем, например, с помощью аппарата сетей Петри. Таким образом, говоря о бизнес-транзакции с точки зрения системного подхода, отметим, что при изучении бизнес-транзакции возможна различная трактовка понятий ее элементов, связей, структуры, целей и других признаков системы, а также внешней для нее среды. Трактовка должна определяться позицией наблюдателя и его целями изучения бизнес-транзакции.

Покажем, что бизнес-транзакцию типа транзакции бизнес-процессов или длительную в основном можно отнести к классу сложных систем. Вообще в теории систем выделяют классы больших и классы сложных систем. В [146] поясняется, что некоторые исследователи используют эти определения как синонимы, например между большими и сложными системами не ставится различий в [151] и у других цитируемых в этом труде авторов. Мы же будем рассматривать большую систему как систему, обладающую большим числом элементов (часто относительно однородных), либо как систему, «которую невозможно исследовать не иначе, как по подсистемам» [152]. Под сложной системой нами понимается система, которая (комбинируя определения из [153], [147], [154]):

- Имеет большое количество связанных и взаимодействующих элементов с высоким количеством разнообразия [155].
- Выполняет сравнительно сложную функцию, направленную на достижение множества целей функционирования.
- Может быть разделена на подсистемы, цель каждой из которых согласуется с общей целью системы.

- Имеет систему управления и интенсивные потоки информации.
- Взаимодействует с внешней средой в условиях воздействия случайных факторов и реагирует на внешние возмущения, сообразуясь с внутренней целью.

В [156] описываются характеристики сложных технических систем, которым бизнес-транзакция также, на наш взгляд, может удовлетворять. Отметим, что сложность бизнес-транзакции подчиняется и определению в [152]: общая цель системы задается множество целей ее участников, а описание и модель транзакции, в зависимости от задач и точки зрения наблюдателя и исследования могут быть различными. Также следует учитывать, что понятие сложности должно определяться субъективной, интуитивной оценкой наблюдателя или группы специалистов, так как перечисленные нами свойства не имеют количественного исчисления.

Шеннон в [157] выделил следующие характеристики сложных систем, которые на наш взгляд, также в полной мере подходят бизнес-транзакциям. Так, бизнес-транзакция может изменяться. Ее элементы, которые, как мы уже отмечали, могут принадлежать различным сферам управления, включаются или исключаются из системы в процессе ее развития или движения (здесь движение рассматривается как процесс последовательного, целенаправленного изменения состояний).

Бизнес-транзакция существует при постоянном наличии и влиянии окружающей среды. Как сложная система она обладает поведением, сложным к интуитивному пониманию, основанном только на поверхностном изучении бизнес-транзакции, ввиду сложных взаимосвязей между ее участниками и элементами. Бизнес-транзакция может обладать тенденцией к изменению характеристик. Шеннон отмечал, что происходит ухудшение характеристик, однако мы не можем принять или отвергнуть это положение для бизнес-транзакций. Из-за того, что бизнес-транзакция представляет собой сложную композицию преимущественно организационных, программных и аппаратных составляющих, на качество ее работы оказывает неоднозначное влияние множество факторов, например, особенности трудовой деятельности человека, процессы приработки и износа аппаратуры, свойства уменьшения (устранения) количества ошибок в программном средстве при его длительной эксплуатации и т.д.

В целом, исследование бизнес-транзакции затруднено ввиду следующих причин:

- Распределенный характер бизнес-транзакции обязывает наблюдателя иметь возможность осуществлять свою деятельность в сферах ответственности различных участников, что зачастую в реальном мире затруднено.

- Сложность, долговременный характер и высокая стоимость каждого исполнения бизнес-транзакции не позволяет проводить ее исследования в естественной среде.
- Сложная неоднородная иерархическая структура бизнес-транзакции не позволяет применять к ней существующие аналитические методы исследования.
- Отсутствие на данный момент средств формализованного представления бизнес-транзакции как целенаправленной системы.

Отметим, что с сущностью бизнес-транзакции инженер (программист, бизнес-аналитик и т.д.) сталкивается каждый раз при проектировании B2B-взаимодействия, сложных многоэтапных бизнес-процессов, взаимодействия множества независимых участников и т.д. Выделение бизнес-транзакций способствует рассмотрению распределенной системы с точки зрения генеральной цели ее существования, а не только как системы последовательной передачи управления, информации, материальных и иных объектов, как она рассматривается в популярных методологиях EPC, IDEF0, BPMN и др. [54].

Все вышесказанное позволяет предполагать, что для исследования бизнес-транзакции, решения задач ее идентификации в распределенной среде, разработки ее структуры и изучения отдельных качеств необходимо использовать методы системного анализа.

Выделим типовые проблемы при работе в области распределенных систем, где необходимо рассмотрение предмета бизнес-транзакции:

- Отсутствует единый центр координации взаимодействия, что не позволяет централизовать контроль над всеми процессами и участниками.
- Сфера управления взаимодействия распределена между несколькими независимыми и/или разнородными участниками.
- Во взаимодействии участвуют процессы разнообразной природы, автоматизированные или ручные, связанные с обработкой объектов материальной или информационной природы и т.д.
- Длительность выполнения взаимодействия превышает пределы существования или работы отдельных систем и элементов, участвующих в управлении: процессов в компьютерных системах, операторов ЭВМ (здесь рассматривается, например, длительность рабочего времени), бумажных носителей информации, материальных ресурсов и т.д.
- Невозможна сколько-нибудь долговременная фиксация частичных результатов взаимодействия – взаимодействие в целом атомарно, оно выполняется либо полностью, либо не выполняется вообще.

- Цель взаимодействия не описывается единичным показателем, а составляет некоторое множество состояний системы.
- Цель взаимодействия определяет точку зрения его проектирования и разработки, являясь ядром этой технологии.
- Целевое состояние каждого участника взаимодействия также описывается множеством возможных состояний, причем отдельные состояния из этого множества невозможно достичь при удовлетворении некоторых конечных состояний другого участника.
- Атомарность взаимодействия возможна за счет описания как множества конечных состояний, так и множества начальных, возврат к которым позволяет инициировать взаимодействие вновь (если предыдущая попытка достижения цели не удалась). Множество конечных состояний образуют цель функционирования системы.

Опишем типовые задачи, решение которых может потребоваться при работе с предметом бизнес-транзакций:

- Выделение сущности бизнес-транзакции в существующей распределенной системе с целью повышения уровня контроля, качества работы системы или преобразования взаимодействия.
- Проектирование и разработка бизнес-транзакций на основе существующих и новых подходов, инструментов и ресурсов распределенной системы.
- Исследование бизнес-транзакции с целью оптимизации в целом или ее отдельных свойств.

Методы решения этих задач лежат в области системных исследований, и при этом одни типы задач могут возникать и при решении других типов. Так выделение бизнес-транзакций в существующей распределенной среде может сводиться к процедурам идентификации систем, где на основании наблюдений строится ряд моделей, описывающих бизнес-транзакцию, и выбирается наилучшая, которая затем кладется в основу исследования, автоматизации или реорганизации взаимодействия или управляющих им систем. Проектирование и разработка бизнес-транзакции, отвечающей заданным характеристиками, обеспечивающей достижение заданного множества целей, с учетом существующих или разрабатываемых ресурсов и инструментов, исследование существующей или проектируемой бизнес-транзакции, ее моделирование, оценка, оптимизация решается методами системного анализа. При этом, ввиду описанных ранее затруднений, принимая во внимания сложность и системность предмета исследования, методы формализованного представления систем должны применяться наравне с методами активизации и использования интуиции специали-

стов (эта классификация методов системного анализа подробно описана в работах под редакцией Волковой В.Н., например в [146], [158] или в [159]).

1.3.4. Сложность оценки надежности бизнес-транзакций

Приведенные нами ранее общие показатели надежности ориентированы в первую очередь на оценку надежности отдельных элементов технических систем. Если же необходимо оценить надежность системы, состоящей из нескольких элементов, то в теории надежности говорят уже о «сложных» системах. Как уже было показано, большинство бизнес-транзакций следует относить к классу сложных систем. В связи с этим должен изменяться и подход к ее оценке надежности. Так, в [153] и [91] утверждается, что использование понятий, заимствованных из теории надежности простых систем, приводит к тому, что учитывается и изучается только факт появления отказов в системе, но не исследуется вопрос их влияния на конечные результаты функционирования системы в целом.

Для предмета сложных систем в особенности изменяется понятие «отказа». Отказ отдельных элементов или подсистем не обязательно влечет отказ всей системы, возможно, изменяются только отдельные характеристики ее функционирования, такое свойство сложных систем можно называть функциональной избыточностью. Числовая характеристика функционирования в [153] называется показателем эффективности системы, однако мы, используя предложения Н.П. Бусленко, в дальнейшем будем отличать это понятие от надежности, следуя терминологии ГОСТ Р ИСО/МЭК 9126-93 [89], и сохраним термин «характеристика функционирования».

В [91] под отказом сложной системы предлагается понимать событие, которое влечет выход характеристик функционирования системы за установленный допустимый предел. Характеристики функционирования системы можно задавать в частном порядке, полагая, например, что к ним относится время наработки на отказ, время ожидания отклика, количество привлекаемых ресурсов и т.д.

Показатели надежности можно оценить, используя теорию надежности «простых» систем, экспериментально или аналитически. Экспериментальный способ предполагает исследование модели или реально-функционирующей системы. Аналитический подход позволяет оценить надежность по структуре и составу системы, исследуя элементы системы и связи между ними. Экспериментальный подход предполагает использование имитационного моделирования, разработки счетчиков и контролирующих устройств, аналитический подход использует математические методы и ограничен сложностью исследуемых систем.

Для расчета надежности отдельных элементов широко используется аппарат распределений вероятностей наработки до отказа, времени ремонта или восстановления. В зависимости от природы элемента, его особенностей функционирования и целей исследователя применяют показательное или нормальное, гамма или треугольное распределение, распределение Вейбулла или Релея, их комбинации ([105, 160, 161, 162, 157] и другие), используют методы интегральных и дифференциальных уравнений. Теоретически, не существует ограничений на использование этого метода и для расчетов структурно-сложных систем. Однако, как отмечают многие авторы, использовать математические модели и расчеты надежности сложных систем трудно, ввиду следующих причин:

- При составлении математической модели возникает погрешность из-за замены функциональных связей между реальными элементами некоторыми логическими соотношениями, учитывающими лишь некоторые состояния системы [163]. Эту погрешность практически невозможно учесть [105], а при замене реальных связей упрощенными логическими соотношениями нарушается принцип эмерджентности системы – рассмотрение элемента или соединения в отрыве от всей системы приводит к невозможности рассмотрения ряда ее свойств, возникающих только в системном окружении.
- Даже для сравнительно простых систем исследование является очень громоздким. Увеличение количества элементов, множественности и сложности их связей экспоненциально увеличивает сложность математической модели. А попытки ее упрощения приводят к возникновению погрешностей оценки [92].
- Итоговые расчетные формулы оценки надежности сложных систем являются очень громоздкими, трудноприменимыми и сложнотиражируемыми [151] со ссылкой на академика Гнеденко), а их применение с разработкой моделирующих алгоритмов и программ трудозатратно [105].

Поэтому среди аналитических методов широкое распространение получили методы структурного расчета. Они исходят из предположения, что любая сложная система состоит из самостоятельных элементов и имеет структуру, сводимую к логическим соотношениям, например, в [99]. Между бизнес-транзакцией как сложной системой в нашем представлении и сложной системой с точки зрения теории надежности существуют определенные отличия. В последней под сложными системами зачастую понимаются системы, имеющие сложную структуру и состоящие из счетного множества элементов. Связи между элементами играют роль каналов передачи информации или непосредственных соединений физической природы.

Таким образом, связи между элементами рассматриваются равноправными и задается условие, что отказ одного элемента приводит к отказу всех элементов, следующих в цепи связей после данного. Конечно, системы технической природы, несомненно, обладают подобными свойствами, и использование описанных допущений является логичным. Они применяются в наиболее популярных методах расчета надежности, которые используют структурные схемы сложных систем: методе преобразования сложных структур, методе декомпозиции, методе минимального сечения, логико-вероятностном методе и т.д. [104, 92, 164].

Использование методов, основанных на анализе структурной схемы системы, также трудоемко для сложных систем: оценка надежности связана с кропотливым неформализуемым трудом по преобразованию схемы или ее отображению в математическую модель (далее мы приведем ряд примеров такой работы). Кроме того, эти методы, как уже отмечалось, не позволяют анализировать систему в целом, упрощают связи между элементами, искусственно ограничивают их возможное поведение и нарушают свойство эмерджентности.

Также отметим, что подобные методы используют упрощенное понятие работоспособности элемента системы, который в любой момент времени может существовать только в двух состояниях: работоспособное и неработоспособное. Работоспособность элемента не зависит от его предыдущих состояний и состояний, связанных с ним других элементов. Элементы системы равноправны и схожи по своему поведению и природе. Эти условия невыполнимы в рамках реально-функционирующей бизнес-транзакции: каждый ее элемент в зависимости от уровня абстракции наблюдателя может являться отдельной функцией, бизнес-процессом, программой-сервисом, устройством или более крупной организационной и технической единицей. Исследовать надежность таких единиц следует с позиции их природы и назначения, что является задачей традиционной теории надежности, теории надежности программных систем, надежности работы человека-оператора и т.д. по всем видам обеспечения ИС. Все это требует комплексного подхода по систематизации и обобщению накопленных знаний и опыта в этих областях и, как уже отмечалось, входит в компетенцию различных специалистов.

Проведение такого комплексного анализа надежности, учитывающего различные аспекты структуры и поведения бизнес-транзакций возможно в рамках расчета функциональной, а не структурной надежности, когда рассчитывается вероятность успешного выполнения функции системой. Функциональный и структурный анализ надежности в целом используют функциональный и структурных подход к анализу систем [165]. В [91] функциональный анализ предлагается осуществлять последова-

тельным расчетом вероятностей работоспособности аппаратуры, надежности математического и программного обеспечения, надежности работы персонала. Итоговый показатель функциональной надежности вычисляется перемножением вероятностей или расчетом условной вероятности. Однако, на наш взгляд, подобный метод обладает существенными недостатком в виде чрезвычайного упрощения и формализации структуры, связей и поведения реальной системы и ее элементов. Фактически, функциональная надежность предполагает аналогичные аналитическим методы расчета, только предметом исследования является выполняемая элементом функция.

Таким образом, по нашей оценке, применение аналитических методов для расчета надежности бизнес-транзакции как сложной системы затруднено и ограниченно, т.к. они не позволяют получать всестороннюю оценку, и оперируют понятием надежности в рамках ряда сугубо технических показателей. Тем не менее, в Приложении Б мы рассмотрели некоторые из этих методов для демонстрации работоспособности предлагаемой методики оценки надежности.

1.3.5. Методы системного анализа надежности бизнес-транзакций

Задача проектирования и разработки бизнес-транзакции может быть решена с использованием методики системного анализа, что предполагает разбиения процесса решения на отдельные этапы. Воспользуемся анализом различных классификаций этапов, приведенных в [152], [154], [153], и сформулируем собственный перечень этапов, наиболее подходящий рассматриваемой предметной области:

1. Выявление и анализ проблемы взаимодействия в распределенной среде, требующий введения предмета бизнес-транзакции. Оценка актуальности и разрешимости проблемы. Анализ существующей системы и/или взаимодействия.
2. Формирование задачи исследования. Определение бизнес-транзакции как системы. Определение позиции наблюдателя, объектов, элементов и связей между ними. Отделение бизнес-транзакции от среды.
3. Определение функций участников, элементов и бизнес-процессов. Определение уровней иерархии элементов, подсистем, бизнес-процессов.
4. Определение целей участников, элементов и бизнес-процессов. Определение начальных и конечных состояний. Оценка взаимозависимости целей, отсечение избыточных и противоречивых. Определение общей цели и начального состояния через множество конечных и начальных состояний.

5. Определение возможностей разработки системы. Оценка существующих ресурсов и инструментов и анализ необходимости разработки новых. Документирование требований к системе.
6. Разработка альтернатив по структуре и составу системы. Разработка альтернативных моделей системы. Оценка и сравнение вариантов. Совмещение отобранных вариантов и формирование проекта окончательного решения. Анализ его параметров. Доработка и исправление.
7. Реализация решения, отдельных элементов, подсистем, функций и процессов. Реализация процессов обмена информацией и управления. Оценка результатов реализации.
8. Внедрение системы. Опытная эксплуатация. Рабочая эксплуатация. Сопровождение и модернизация системы.

Как уже отмечалось, в целом разработка бизнес-транзакции представима в качестве процесса синтеза системы по заданному поведению и набору элементов, а отдельные этапы и подэтапы могут требовать использования различных методов системного анализа. Одним из наиболее важных этапов, на наш взгляд, является оценка проекта окончательного решения. От результатов этапа зависит качество разработанной по проекту бизнес-транзакции: ее функциональность, работоспособность, эффективность надежность и т.д. Этот этап может применяться и в случаях исследования и улучшения существующих взаимодействий. Рассмотрим процесс анализа надежности проектируемой бизнес-транзакции. Предположим, что оценка надежности ведется безотносительно к более крупному процессу разработки, проектирования или комплексного исследования бизнес-транзакции, т.е. положим, что задача анализа надежности сформулирована в частном порядке по существующей или проектируемой системе.

Учитывая вышесказанное и рекомендации по управлению надежностью проектируемых изделий из [105], [109], [91] выделим следующие этапы алгоритма оценки надежности:

1. Формулирование и анализ проблемы надежности системы. Определение взаимосвязи надежности с другими характеристиками. Оценка разрешимости проблемы.
2. Формирование целей и задач исследования. Определение целевых качественных и количественных показателей надежности. Определение нормативных значений этих показателей.
3. Анализ структуры и состава системы. Определение связей и функций участников, элементов и процессов.

4. Определение возможных методов оценки надежности, используемых показателей и метрик. Анализ вариантов. Выработка конечной технологии оценки.
5. Оценка надежности системы по выработанной технологии. Оценка результатов.

Не исключается возможность итеративного повторения перечисленных этапов в случае, когда результаты оценки надежности не удовлетворяют целевым показателям. Проведем адаптацию перечисленных этапов оценки надежности к предмету бизнес-транзакции.

1.3.6. Использование имитационного моделирования для оценки надежности проектируемой бизнес-транзакции

Считается, что основные работы по обеспечению надежности любого разрабатываемого изделия проводятся на этапе его проектирования [105], [162], [160], [109], [151], [98], [99]. Вне зависимости от природы изделия во время этого этапа определяют требования к надежности, ее показатели и их нормы. Заданный уровень надежности влияет на все конструктивные решения и параметры, принятые по системе, на ее структуру и архитектуру, номенклатуру элементов, режимы будущей работы и т.д.

В. В. Шураков в [109], Ю.А. Соколов (и др.) в [99] выделяют три основных задачи, решаемых на этапе проектирования: обоснование требований к надежности, проектирование системы с заданным уровнем надежности и количественная оценка надежности предпочтительного варианта реализации. Похожего мнения придерживается Дружинин Г.В. в [105].

Обоснование требований к надежности, выбор показателей и их норм может проводится двумя путями: экспертным заключением, которое основывается на опыте, интуиции эксперта и требованиях внешней среды, или решением оптимизационной задачи, состоящей в нахождении максимума некоторой функции от вариантов реализации системы, затрат на ее реализацию, функционирование и сопровождение, количества и качества элементов и т.д. Задание оптимального уровня надежности возможно только в случае, когда [166]:

- Выходной результат функционирования системы измерим в тех же единицах, что и стоимость ее создания и обеспечения надежности.
- Известны параметры надежности, функционирования и стоимости всех элементов системы.
- Определены структура системы и принципы ее функционирования.

Пример подобной задачи был поставлен, например, в [91] и [99].

Ясно, что любые расчеты надежности во время проектирования носят в основном теоретический характер: проводить испытания на надежность для всех вариантов реализации невозможно или экономически нецелесообразно [105]. Но, несмотря на то, что на этапе создания и эксплуатации системы расчет надежности можно проводить по результатам испытаний и экспериментов, такой расчет носит, как правило, характер констатации [91], [92]. Это позволяет выявить слабые стороны системы, ошибки проектирования, особенности влияния на систему различных факторов и выработать меры по повышению ее надежности. Однако внести существенные изменения в уровень надежности, также как и, например, изменить архитектуру системы, на последних этапах разработки системы невозможно.

Вышесказанное позволяет сделать вывод, что этап проектирования – важный период для определения надежности будущей системы. При этом исследователь обладает возможностью только расчетов с использованием справочной литературы, средств проектирования и моделирования, а получаемые на практике данные ограничены исследованием свойств отдельных элементов и подсистем, которые, исходя из принципов системного анализа, не позволяют достоверно судить о поведении системы в целом.

Выделяют два способа расчета надежности системы при ее проектировании [105]: с составлением математической модели функционирования системы и без ее составления по структурной схеме системы. Использование математических моделей, как уже отмечалось, плохо подходит для сложных систем: исследователю приходится находить баланс между точностью расчетной модели, ее простотой и обеспеченностью исходными данными. В результате чего теряется требуемый уровень достоверности и адекватности. Методы с использованием схемы системы опираются на методы структурного и функционального анализа, что, как мы уже отмечали, влечет трудоемкую работу по преобразованию схем, расчетам надежности различных подсистем и функциональных «срезов» системы, что, в итоге, отрицательно сказывается на экономической целесообразности и достоверности расчетов.

Однако среди методов системного анализа существуют такие, которые, с одной стороны, предоставляют возможности анализа сложной системы с учетом ее сложных взаимосвязей и поведенческих аспектов, с другой стороны не требуют больших материальных, трудовых и временных затрат, например, метод имитационного моделирования систем. Он используются в том случае, когда невозможно получить аналитическое решение задач теории надежности [95, 91, 92, 147, 163] или по причинам, например, отсутствия изучаемого объекта в действительности (например, при проек-

тировании системы), сложности/невозможности постановки эксперимента на реальном объекте и др. (описанным, например, в [157]).

Задачей имитационного моделирования является имитация функционирования системы с целью исследования ее поведения и свойств. Используя в качестве основы определение из [146], скажем, что имитационное моделирование – это метод моделирования, реализуемого с помощью набора математических средств, специальных программ и технологий программирования, позволяющий провести исследование структуры и функций сложной системы. Под имитационной моделью будем понимать модель, разработанную с помощью специальных инструментов и средств, позволяющих имитировать поведение какого-либо объекта.

Выделим следующие преимущества использования имитационных моделей для оценки надежности проектируемых сложных систем [92, 155, 157, 163, 165]:

- Имитационная модель обладает бóльшей информативностью и понятностью для исследователя и специалиста в предметной области. Поэтому модель легко оценить в точки зрения адекватности и достоверности.
- Имитационная модель позволяет учитывать динамику отказов элементов системы, а также выявлять последствия подобных отказов в масштабе всей системы.
- Имитируя различные реальные ситуации, исследователь может изучать показатели общей надежности и выявить такой состав и структуру системы, позволяющие максимизировать эти показатели.

Опишем последовательность моделирования с целью изучения надежности проектируемой бизнес-транзакции:

1. определение целей моделирования;
2. выбор показателей надежности, которые необходимо определить или оптимизировать;
3. формализация моделируемых процессов с учетом целей и выбранных показателей;
4. разработка моделирующего алгоритма;
5. реализация алгоритма и анализ результатов.

Имитационное моделирование бизнес-транзакции позволит выявить характеристики ее работы и определить основные показатели надежности, перечисленные в табл. 4. Полученные показатели можно сравнить с требуемыми и сделать вывод о достижении или не достижении транзакцией необходимого уровня надежности. Если требуемые показатели надежности не достигнуты, то исследователю необходимо выявить пути повышения надежности объекта и внести изменения в моделирующие ал-

горитмы, т.е. повторять пункты 4 и 5 последовательности до тех пор, пока не будет получен необходимый результат.

При оценке надежности системы методом имитационного моделирования можно получить большое количество информации о работе бизнес-транзакции. Очевидно, что исследователя в меньшей степени интересует смысл тех выходных данных, которые транзакция создала в ответ на входные данные: для каждой бизнес-транзакции набор таких данных могут существенно отличаться, что не позволяет разработать универсальный механизм оценки надежности. То есть информация, полученная в результате моделирования и позволяющая оценивать надежность, должна:

- во-первых, не зависеть от природы, целей и задач бизнес-транзакции;
- во-вторых, являться универсальным измерителем, позволяющим проводить сравнения надежности различных систем;
- в-третьих, с целью оптимизации аналитической работы исследователя и программных систем, получаемая информация должна использоваться при расчете как можно большего количества показателей надежности;
- в-четвертых, легко поддаваться измерению и вычислительным операциям.

Таким универсальным показателем, на наш взгляд, является количество отказов. При точной формулировке толкование отказа может не зависеть от области приложения бизнес-транзакции, количество отказов легко может измеряться как процессе работы самой исследуемой системой, так и с помощью внешнего наблюдателя, и эта статистика повсеместно используется при расчете показателей надежности бизнес-транзакции (см. табл. 4).

Важным остается вопрос о том, каким образом исследователь может выявить пути повышения надежности системы. Исходя из общих и системных свойств бизнес-транзакции ясно, что качество ее функционирования определяется элементами, ее структурой и их взаимодействием, а фактором влияния на надежность технической связи между элементами мы предлагаем пренебречь: по сравнению со временем выполнения операций время передачи данных между ними невелико, а отказ во время передачи можно моделировать с помощью отдельного перехода.

Однако это заключение не позволяет выявлять зависимость между общей надежностью транзакции и надежностью ее отдельных элементов, хотя мы предположили, что надежность транзакции есть некая функция от надежности ее составляющих. Т.е. имеет смысл отыскать процедуру выявления тех подпроцессов, которые влияют на надежность существеннее других, и минимизации уровня отказа этих операций.

Выделим следующие задачи, который потенциально может решать метод имитационного моделирования при анализе надежности бизнес-транзакции:

1. Оценка общей надежности системы при заданных параметрах. В этом случае происходит подстановка в модель существующих параметров функционирования операций и отказов подсистем. Моделирование позволяет определить особенности функционирования и надежности всей системы.
2. Сравнение надежности альтернативных вариантов реализации системы. Эта задача является расширением предыдущей, но проводится анализ с несколькими наборами параметров или моделей системы с целью поиска оптимального варианта реализации проекта.
3. Анализ чувствительности. Предполагается, что одни операции больше влияют на надежность системы, чем другие. Степень их влияние и сравнение влияний разных операций друг с другом позволяет выявить анализ чувствительности.
4. Оптимизация надежности. Задача состоит в варьировании уровней надежности операций в заданных пределах с целью отыскания наиболее надежного режима функционирования бизнес-транзакции.

Первая задача легко решается с помощью статистического анализа результатов имитационных прогонов, когда могут быть получены достоверные интервальные или точечные оценки показателей надежности.

Вторая задача хорошо подходит для анализа различных вариантов реализации системы, например, различных структур, порядка взаимодействия, для нагрузочного тестирования и т.д. Оценки универсальных показателей надежности позволяют сравнивать варианты реализации между собой и выбирать более эффективный.

Третья задача предполагает широкое использование методов планирования экспериментов и, например, регрессионного анализа для истолкования результатов. Выявленные закономерности между поведением подсистем и воздействием их на общую надежность, возможно, могли бы способствовать разработке некоторого универсального регрессионного уравнения, которое описывало бы зависимость между отказами всей бизнес-транзакции и отказами ее отдельных подсистем. В основе предпосылок к созданию классической модели множественной регрессии лежит то, что факторы, включаемые в уравнение, должны быть количественно измеримы и независимы [167]. Как уже отмечалось, количество отказов или факт наступления отказа является статистикой, наиболее часто используемой при расчете показателей надежности, поэтому, возможно, либо этот показатель, либо зависимый от него могут быть кандидатами в факторы уравнения регрессии. Рассмотрим простейшую

структуру бизнес-транзакции, когда *n* сервисов располагаются в линейной цепочке передачи данных друг к другу. Формально такая структура может соответствовать линейному уравнению регрессии. В случае отказа любого сервиса при имитации невозможно установить откажут ли следующие в цепи после данного, а при отказе любого сервиса наступает отказ всей транзакции. Т.е. все факторы зависимы, что противоречит требованиям регрессионного анализа. С другой стороны, можно воспользоваться относительным показателем, который не зависит от предыдущих элементов, например, частотой наступления отказов, где учитываются отказы и попытки выполнения текущего элемента. В этом случае требования выполняются, однако возникает вопрос о выборе формы регрессионного уравнения – ввиду того, что количество участников транзакции может быть велико, а зависимость между ними может описываться сложными структурами или процессами, выходящими за пределы ответственности бизнес-транзакции, то уравнение может быть сложным, а величина случайной ошибки недопустимо большой. Таким образом, попытка формализовать результаты имитационного моделирования приводит к появлению проблем, типичных для аналитических методов расчета надежности, которые слабоприменимы с учетом специфики бизнес-транзакции.

Четвертая задача может предполагать минимизацию уровня отказов исследуемой системы путем влияния на параметры надежности ее подсистем. Логично предположить, что в большинстве случаев надежность (выраженная в количестве отказов) всей бизнес-транзакции будет находиться в прямой пропорциональной зависимости от надежности входящих в нее подсистем, т.е. чем выше надежность подсистем, тем выше надежность и системы в целом вне зависимости от структуры и характера связей между составляющими. Таким образом, четвертая задача может рассматриваться как задача повышения надежности системы путем повышения надежности ее подсистем. Ввиду того, что чаще всего параметры функционирования подсистем в бизнес-транзакции, которые влияют на их надежность, либо не могут изменяться вовсе (например, бизнес-транзакция использует сторонний сервис, надежность которого находится вне пределов контроля бизнес-транзакции), либо могут изменяться дискретно с заданным шагом (например, путем дублирования стороннего сервиса), и в силу описанных ранее ограничений по природе бизнес-транзакции такие часто используемые в имитационном моделировании математические методы многокритериальной оптимизации, как, например, методика анализа поверхности отклика, тоже не могут быть применены.

Перечисленные задачи могут определять цели моделирования, поэтому для проведения исследования надежности необходимо разработать технологию форма-

лизации моделируемых процессов, их имитационной реализации и показатели, оценивающие их надежность.

Выводы по главе 1

Бизнес-процесс, совместно выполняемый несколькими взаимодействующими ПОИС, часто должен быть либо выполненным полностью, либо не выполненным вообще. Этим свойством он повторяет свойство атомарности системных транзакций из теории баз данных, и поэтому может быть назван «бизнес-транзакцией». Однако бизнес-транзакция по своей природе намного сложнее. Например, она выполняется длительное время, может быть необратима, к ней неприменимы стандартные транзакционные свойства и средства управления и т.д.

Ввиду распространенности и перспективности сервис-ориентированного подхода к интеграции бизнес-транзакция будет рассматриваться выполняющейся в среде взаимодействующих сетевых служб.

Мы предлагаем понимать под бизнес-транзакцией согласованное изменение состояния нескольких участников взаимодействия, что на практике в сервис-ориентированных системах выражается как неделимая последовательность операций (акт взаимодействия) взаимодействующих сервисов, выполняющих совместный бизнес-процесс. Бизнес-транзакция представляет собой сложную систему, функционирующую в распределенной среде, то для анализа ее надежности может быть стандартные методы теории надежности могут быть недостаточно эффективными. Поэтому был предложен расчет методом имитационного моделирования и определен круг задач, потенциально решаемых этим методом.

Представление процесса взаимодействия в распределенных системах в виде бизнес-транзакции позволяет объединить различные подходы к анализу надежности систем, абстрагироваться от их различной природы, а также использовать методы системного анализа. Сервис-ориентированная среда выполнения позволяет использовать методы повышения надежности, недоступные ранее в исключительно программных и организационных системах, например, автоматический поиск дублирующих сервисов и использование их в случае отказа основных подсистем.

Все это открывает широкие возможности для интеграции современных информационных технологий, теории распределенного взаимодействия и природы бизнес-транзакции в целях повышения качества и надежности работы сложных систем.

Эти выводы были получены в первой главе с учетом следующих результатов:

1. Рассмотрено понятие бизнес-процесса, процессного подхода к управлению организацией и понятие процессно-ориентированной информационной системы. Рассмотрены причины необходимости интеграции приложений предприятия. Интегрированная система, приобретающая новые функции, является распределенной программной системой. Введено понятие «связывания» как стиля организации взаимодействия между интегрируемыми системами. Выделены типы связывания. Показано, что развитие теории веб-служб привело к появлению новой идеологии построения информационной инфраструктуры компании – сервис-ориентированной архитектуры (СОА). Дано определение, свойства и ключевые особенности СОА.
2. Рассмотрены стандарты моделирования СОА и бизнес-процессов. Показано, что существует технология, которая в отличие от большинства стандартов позволяет строить точные формальные модели системы – сети Петри. Описаны преимущества сетей Петри. Обоснована применимость сетей Петри для моделирования бизнес-процессов и сервис-ориентированных систем.
3. Рассмотрено понятие качества программной системы. Выделена надежность как одна из ключевых проблем интегрированной ПОИС. Рассмотрены подходы к определению надежности. Описаны показатели надежности технических систем. Приведены причины необходимости особого подхода к оценке надежности программных систем. Рассмотрены определение и атрибуты надежности: стабильность, отказоустойчивости и восстанавливаемость. Перечислены показатели надежности программных систем. Предположено, что сервис-ориентированная концепция позволяет обеспечить высокий уровень восстанавливаемости и отказоустойчивости, и, поэтому, стабильности системы. Перечислены стандарты для веб-служб, позволяющие обеспечить определенный уровень надежности сервис-ориентированной системы.
4. Проанализированы существующие методы анализа надежности и ограничения, которые на них накладывает предмет бизнес-транзакции и ее свойства. Бизнес-транзакция рассмотрена с трех точек зрения надежности: как транзакция, композиция сервисов и бизнес-процессов.
5. Дано определение системной («плоской») транзакции в базах данных, описаны ее свойства и технологии управлениями ею. Показано, что некоторые бизнес-процессы принимают черты системной транзакции. Подоб-

ные длительные бизнес-процессы с признаками атомарности были названы «бизнес-транзакциями». Показаны идеологические соответствия между «бизнес-транзакциями» и каноническими СУБД-транзакциями.

6. Предложено определение бизнес-транзакции. Описаны характерные черты бизнес-транзакции и требования для ее поддержки.
7. Предложены свойства бизнес-транзакции и показаны отличия от системных транзакций. Показано, что к бизнес-транзакциям невозможно применить традиционные подходы к управлению транзакциями. Введено понятие компенсации как семантического «отката» бизнес-транзакции.
8. Предложено рассматривать бизнес-транзакцию как целостную систему. Описана ее структура, связи, элементы. Выделены типовые проблемы в области проектирования и разработки распределенных систем, где будет полезно использование бизнес-транзакции как объекта изучения. Показано, что бизнес-транзакция является сложной системой. Предложены этапы системного исследования ее в целом и ее надежности. Этот системный базис лежит в основе предложенной далее методики анализа надежности бизнес-транзакции.
9. Основные работы по обеспечению надежности должны проводиться на этапе проектирования системы. На основании проведенного анализа существующих методов надежности предложено для оценки разрабатываемой бизнес-транзакции воспользоваться методом имитационного моделирования. Описаны преимущества этого метода по сравнению с аналитическими подходами. Предложен универсальный показатель надежности, который лежит в основе расчетов большинства других. Предложен и проанализирован набор задач в области бизнес-транзакций, решение которых можно достичь методом имитационного моделирования.

Бизнес-транзакция, исполняемая взаимодействующими ПОИС, может вовлекать в свою работу различные средства, иметь сложную логику обращения к ним и обладать непростой внутренней структурой. Поэтому необходимо использовать принципы системного подхода, которые позволят отделить бизнес-транзакцию от внешней среды и провести ее изучение как сложной, самостоятельной системы.

Бизнес-транзакция функционирует в экономической среде и является, по сути, бизнес-процессом, на который налагается ряд ограничений. Его выполнение предполагает совместную работу нескольких самостоятельных участников, интерфейсы вызова которых могут быть оформлены в виде сетевых служб. В совокупности эти особенности не позволяют оценивать надежности бизнес-транзакции просто как про-

граммы, или системной транзакции или бизнес-процесса в отдельности, а требуют создания комплексного подхода.

В связи с этим оценка ее надежности представляется трудоемкой задачей, предполагающей, как минимум, последовательный расчет надежности каждой вовлекаемой операции (а операция выполняется с привлечением разных видов обеспечения АИС, каждое из которых требует применения специфических методов расчета надежности) с учетом структуры и последовательности их вызовов. Другим методом может выступать метод имитационного моделирования, который хорошо применим к анализу таких сложным систем и явлений как бизнес-транзакция.

Поэтому задачами 2-ой главы являются:

- Рассмотреть причины сложности оценки надежности бизнес-транзакции. Выработать комплексный подход к оценке надежности, предполагающей, что бизнес-транзакция имеет тройственную природу: является бизнес-процессом, транзакцией и композицией взаимодействующих программ.
- Разработать набор универсальных показателей, позволяющих оценивать надежность бизнес-транзакций.
- Определить средство для моделирования и формального описания сервис-ориентированных систем и бизнес-процессов.
- Дать определение элементов, структуры, связей бизнес-транзакции, выполняемой в сервис-ориентированной среде, в терминах выбранного средства моделирования.
- Определить бизнес-транзакцию в целом и условия ее надежности в терминах выбранного средства моделирования. Определить типовые структуры оркестровки и хореографии участников бизнес-транзакции, ее восстанавливаемые и компенсируемые состояния.
- Описать способы расчета показателей надежности моделируемой бизнес-транзакции.
- Описать методику расчета надежности бизнес-транзакции.
- Описать программный комплекс, позволяющий частично автоматизировать методику исследования надежности.

2. Методическое и программное обеспечение исследования надежности бизнес-транзакций

2.1. Тройственная природа бизнес-транзакции в сервис-ориентированной среде

Идея бизнес-транзакций легко применяется в рамках среды СОА. В этом случае участниками транзакции становятся сервисы, выполняющие отдельные функции, которые можно представить в виде атомарных или длительных транзакций. И с точки зрения процессного подхода бизнес-процессом может быть как отдельный сервис, так и бизнес-транзакция целиком. И, в отличие от других технологий интеграции, СОА позволяет реализовывать как все свойства, так и все типы бизнес-транзакций: атомарные за счет функций сервисов, длительные за счет их композиции и оркестровки, и транзакции бизнес-процессов за счет хореографии. Эта архитектура обладает всеми техническими и методологическими возможностями по организации длительного, слабосвязного, но в то же время надежного взаимодействия участников с целью реализации общего бизнес-процесса.

В [168] под надежностью в бизнес-транзакции понимается то, что процесс выполняется так, как запланировано, каждый участник демонстрирует согласованное поведение, а транзакция завершается в расчетное время и в расчетном месте. Это определение в целом соответствует определению надежности, данных в стандартах ГОСТ Р ИСО/МЭК 9126-93 [89] и ISO/IEC 9126, однако не является ни исчерпывающим, ни охватывает сложную природу бизнес-транзакции в сервис-ориентированной среде.

Как уже отмечалось, надежность программной системы в основном определяется тремя показателями: стабильностью, отказоустойчивостью и восстанавливаемостью. Для бизнес-транзакции стабильность является слабоконтролируемым параметром, который зависит от стабильности выполняющих транзакцию участников. Наличие множества участников в целом ослабляет стабильность бизнес-транзакции по сравнению с атомарными аналогами или распределенными ACID-транзакциями [168]. Но в рамках СОА особое значение приобретают факторы отказоустойчивости и восстанавливаемости, влияние которых, как уже отмечалось, имеет особое значение благодаря врожденным свойствам сервис-ориентированных сред [169]. С этой точки зрения бизнес-транзакция может обладать запасом надежности, уровень которого пропорционален количеству входящих в среду функционально-однородных сервисов – чем больше количество взаимозаменяемых частей бизнес-процесса существует, тем ниже вероятность отказа его целиком. Таким образом, реализация биз-

нес-транзакций в сервис-ориентированной среде обеспечивает ей ранее определенные преимущества и предполагает оперирование понятием надежности уже в области сервисов и их композиции [170].

Представление бизнес-процесса в виде транзакции, выполняемой композицией сервисов, позволяет нам оперировать понятием надежности сразу в трех плоскостях: надежностью композиции сервисов, надежностью бизнес-процесса и надежностью транзакции. Т.е. с технической точки зрения, оценивая показатели надежности программной системы, с точки зрения «бизнеса» оценивая стоимость, время и достоверность ее результатов, и, с функциональной точки зрения, рассматривая его как надежный способ изменения состояния интегрированной бизнес-системы, где не допускается возможность такого частичного или неполного перехода. Для этого будем рассматривать бизнес-транзакцию в сервис-ориентированной среде в разрезе трех измерений надежности: как транзакцию, как композицию сервисов и как бизнес-процесс.

Подобная тройственная природа бизнес-транзакции неявно выделяется в работах Б. Кратца (англ. Benedikt Kratz) и М. Папазоглоу (англ. Mike P Papazoglou) в [120], [123], [171] и [172]. Однако авторы рассматривали характеристики и свойства бизнес-транзакции, вопросы реализации и управления в сервис-ориентированной среде, сосредоточились на формальном описании ее модели и не уделили внимания проблемам надежности.

Рассмотрим сущность транзакции как для процесса, так и для композиции служб. Вопросами придания бизнес-процессам ACID-свойств (в первую очередь, атомарности) СУБД-транзакций и их расширения активно обсуждались начиная с появления WFMS-систем. Например, в [136], [173], [174], [175], [176], [177]. Расширения ACID-свойств на среду процессно-ориентированных информационных систем рассматривались ранее. Эти задачи ослабления были продиктованы невозможностью вписать в требования плоских транзакций всех свойств подобных систем и реализуемых ими процессов. В итоге был разработан набор подходов к реализации иерархической, цепочечной или многоуровневой модели транзакций, позволяющих с некоторыми допущениями поддерживать присущую системным транзакциям надежность [20].

Однако среди всех ACID-свойств наиболее сложным для реализации в бизнес-транзакции является свойство изолированности [17]. В общем случае, в отличие от системных транзакций, для бизнес-транзакций невозможно гарантировать полную изолированность даже случае сериализации всех происходящих транзакций в некоторой системе. Это связано с тем, что бизнес-транзакция может пересекать пределы

ответственности отдельной системы управления или организационной структуры, где любые изменения, произведенные этой транзакций, могут повлиять на результаты выполнения других операций, происходящих вне затрагиваемой транзакции сферы. В [17] предложены простые архитектурные шаблоны, позволяющие достигать изолированности бизнес-транзакции, но они, как и поддержка уровней изолированности системных транзакций в СУБД рассчитаны только на единый центр управления и контроля. Таким образом, предмет поддержки изолированности бизнес-транзакций требует отдельного исследования и не будет рассматриваться в данной работе.

Проблема композиции служб, на наш взгляд, является наиболее тщательно изучаемой во всей теории СОА и веб-сервисов ввиду того, что возможность такой слабосвязной композиции является ключевым отличием сервиса от других технологий интеграции и представления распределенных объектов. Эти вопросы раскрываются в отраслевых отчетах и диссертациях в [178], [179], [180], [181], [182], [183], [184], [185], [186]. Атомарность бизнес-транзакции с точки зрения бизнес-среды предполагает и атомарность процессов, протекающих в композиции сервисов, поэтому отметим, что транзакционные свойства интегрированных служб и их надежность исследовались в [187], [188], [189], [190], [144].

В [191] предлагается модель оценки качества бизнес-процесса, в целом являющаяся укороченной версией модели качества, предложенной в стандартах ГОСТ Р ИСО/МЭК 9126-93 и ISO/IEC 9126, и предлагающая 4 характеристики качества процесса: сопровождаемость, практичность, надежность, функциональные возможности. Однако для понятия надежности используются только два свойства: отказоустойчивость и восстанавливаемость. Отказоустойчивость трактуется как способность процесса избегать ошибок, допускаемых пользователями и участниками, а восстанавливаемость обеспечивает возможность возобновления работы процесса после сбоя с некоторой, ранее определенной позиции. В [192] надежность процесса напрямую переносится к надежности сервиса, его обеспечивающего, т.е. автор сводит надежность к стандартным параметрам QoS сервиса: безопасности, надежности и производительности.

Термин QoS (англ. Quality of Service – качество обслуживания) изначально появился в теории электрических, компьютерных сетей, а также промежуточного ПО [193], и, согласно, [194] качество обслуживания – «это совокупность характеристик услуги, которые имеют отношение к ее возможности удовлетворять установленные и предполагаемые потребности пользователя». Соглашение о качестве обслуживания может иметь фиксированную форму в виде официального документа между постав-

щиком и потребителем сервиса (сервис-ориентированной среде роль такого документа может выполнять сервисный контракт между провайдером и клиентом службы). Этот документ часто называют «соглашение об уровне обслуживания» (англ. SLA) [195]. Например, в [196] и [195] надежность отнесена к SLA, но в [195] надежность является лишь частью компонента SLA «уровень услуги» вместе с доступностью и производительностью. Что не противоречит стандарту [194], где надежность относится к одному из параметров QoS, именно поэтому такой подход к оценке надежности сервисов и их композиций как составляющей QoS является достаточно популярным.

В [197], [187] и [197] отмечается, что QoS композиции сервисов в целом трудно управлять ввиду слабой связанности и распределенности границ ответственности за весь процесс между несколькими участниками, а разработанные ранее методики для оценки QoS нераспределенного, статичного программного обеспечения в рамках сервис-ориентированной среды не являются валидными.

Но официальных стандартов, покрывающих область QoS для веб-служб, не существует, потому разные исследователи предлагают собственные подходы к реализации QoS для сервисов и их композиций: [198], [199], [200], [201], [202]. В [201] для атомарного сервиса под надежностью понимается вероятность того, что на запрос будет получен корректный ответ в течение периода ожидания, максимум которого определен в метаданных сервиса или она может быть измерена как отношение успешных вызовов сервиса к общему числу вызовов. В [203] в зависимости от типа отказа модели оценки надежности программного обеспечения делится на 4 типа: время между сбоями, количество сбоев, модель подмешивания ошибок, модель на основе диапазона входных значений. Первые две модели используются при тестировании и отладке программ и позволяют предполагать, что надежность ПО растет с уменьшением количества ошибок, т.е. называются моделями роста надежности. Две другие модели не связаны с процессом устранения ошибок, они обычно используются при заключительном тестировании программы для оценки финального уровня надежности. Наиболее полная оценка надежности сервиса была дана в [204]. По мнению авторов, понятие надежности включает в себя 2 группы: функциональную и нефункциональную надежность. Функциональная содержит параметры корректности, отказоустойчивости, тестируемости. Нефункциональная определяет уровень интероперабельности и своевременности, который в свою очередь, включает доступность и производительность.

Очевидные проблемы возникают при попытке оценить надежность композиции сервисов. Во-первых, часть метрик надежности является качественной по своей при-

роде (например, интероперабельность), во-вторых, авторы, предлагая отдельные показатели надежности атомарности сервиса, оставляют открытым вопрос расчетов этих параметров для композиции. В-третьих, сервисы могут не быть связаны в последовательные цепочки вызовов, что усложняет методику расчета суммарной надежности, а их взаимодействие может управляться сложным комплексом бизнес-правил и ограничений.

Поэтому для оценки надежности набора служб рассматривают типовые шаблоны взаимодействия бизнес-процессов (учитывая, что в зависимости от точки зрения сервис является бизнес-процессом и наоборот), разработанные еще для WfMS-систем [205] или веб-служб [206]. Например, исследование подходов к QoS для бизнес- и workflow-процессов проводится в работах Х. Кардосо (исп. Jorge Cardoso), например, в [193], [207] и [208]. Объединяя несколько подходов автор вырабатывает собственную номенклатуру параметров, определяющих QoS workflow/бизнес-процесса: время, стоимость, надежность и точность. Время включает различные темпоральные характеристики вида время отклика, время обработки, время задержки и пр. Стоимость выражает финансово-стоимостные параметры процесса. Точность определяется качеством готового продукта или оказанной услуги. Надежность определяется Кардосо с точки зрения двух подходов: дискретно и непрерывно-временного. Дискретный подход определяет надежность как соотношение между успешными и неуспешными состояниями завершения процесса. Непрерывный подход оценивает надежность в терминах периодов времени между сбоями или количестве сбоев за определенный период времени. Для композиции сервисов надежность определяется произведением значений надежности все участников, однако в зависимости от типа связи и ветвления Кардосо предлагает различные вариации технологии расчета совокупной надежности. В [209] QoS атрибуты композиции сервиса включают такие атрибуты как: время отклика, затраты, надежность и доступность. При этом надежность, как и в [201], также определяется произведением значений надежности всех входящих в композицию сервисов.

Подытоживая вышесказанное, определим *понятие надежности бизнес-транзакции как «совокупность свойств, относящихся к способности бизнес-транзакции выполняться и ее участникам согласованно ее выполнять с должным уровнем качества при установленных условиях за установленный период времени».* Для оценки уровня качества бизнес-транзакции воспользуемся критериями из [89], [113] и отнесем к показателям качества функциональные возможности, эффективность, мобильность и сопровождаемость. К атрибутам надежности отнесем группу транзакционных атрибутов, включающих атомарность, согласованность, долговеч-

ность и группу атрибутов надежность программных систем: отказоустойчивость, восстанавливаемость и стабильность:

- Атомарность бизнес-транзакции предполагает, что она должна либо выполниться целиком, либо не выполниться вообще.
- Согласованность, следуя определению, предполагает переход бизнес-системы в процессе бизнес-транзакции из одного согласованного состояния в другое.
- Долговечность означает надежную фиксацию измененного состояния среды всеми участниками.
- Восстанавливаемость означает возможность восстанавливать работу транзакции в случае отказа как любого из участников, так программного и аппаратного обеспечения.
- Стабильность определяется совокупностью свойств, характеризующих частоту отказов как любого из участников транзакции, так программного и аппаратного обеспечения.
- Отказоустойчивость бизнес-транзакции предполагает ее способность поддерживать определенный уровень качества функционирования в случаях программных ошибок или ошибок пользователей при ручных операциях.

Таким образом, приведенные ранее примеры оценок надежности атомарного сервиса (например, в [201]) являются оценками стабильности бизнес-транзакции. Т.е. вероятность отказов, или количество отказов на единицу времени, или время между отказами позволяют оценивать как стабильность отдельных служб, так и при использовании, например, методики Кардосо [193], стабильность транзакции целиком.

Отметим, что свойство согласованности надежной бизнес-транзакции проистекает из самого ее определения и должно решаться на фазе ее идейного планирования. Т.е. участники вступают во взаимодействия только в том случае, когда соглашаются совместно принять все изменения, произведенные этой транзакцией. Свойство долговечности, аналогично, не зависит от структуры и характера исполнения транзакции. И, в случае успешного ее завершения, длительность гарантируется каждым из участников. Однако свойство длительности может быть нарушено, например, вследствие нарушения изолированности транзакции и влияния последующих компенсирующих действий, инициированных другой операцией. Что опять же, выходит за пределы ответственности самой транзакции.

Поэтому выделим понятие «внутренне-надежной бизнес-транзакции», надежность которой не зависит от поведения участников после ее завершения. *Внутренне-надежной будем называть такую бизнес-транзакцию, которая способна вместе со своими участниками во время выполнения поддерживать на определенном уровне*

значения атрибутов, относящихся к ее способности выполняться с должным уровнем качества при установленных условиях за установленный период времени. К атрибутам надежности внутренне-надежной бизнес-транзакции относятся атомарность, восстанавливаемость, отказоустойчивость и стабильность [210]. Естественным образом можно предположить, что внутренняя надежность бизнес-транзакции или ее отдельные характеристики могут быть каким-либо образом предсказаны по ее структуре, а окончательная оценка может быть дана по результатам тестирования или моделирования.

Таким образом, предлагаемая нами система показателей надежности бизнес-транзакции выглядит так (см. рис. 1):

Рисунок 1 – Система показателей надежности бизнес-транзакции

2.2. Показатели надежности бизнес-транзакций

Дадим дополнительную классификацию отказов. Программный отказ связан с нарушением работоспособности по причинам ошибок в программном, информационном и математическом обеспечении. Аппаратный отказ вызывается проблемами с оборудованием. Организационный отказ связан с нарушениями в системе «человек-техника», действием или бездействием человека-исполнителя, повлекшие за собой нарушение хода выполнения бизнес-транзакции. Программный и аппаратный отказ вместе составляют группу технических отказов.

Под отказом бизнес-транзакции будем понимать событие, заключающееся в нарушении хода выполнения бизнес-транзакции. Авария – отказ, повлекший за собой невозможность дальнейшего выполнения или компенсации бизнес-транзакции. Отказ и авария операции определяются таким же образом.

Восстановленным отказом назовем такой отказ, последствия которого были успешно устранены в ходе текущего сеанса выполнения бизнес-транзакции. Иначе говоря, нарушение хода выполнения из-за данного отказа было устранено без прекращения работы бизнес-транзакции по текущей задаче. Например, невозможность выполнить функцию одним сервисом влечет отказ выполнения, однако оркестратор может передать выполнение этой функции другому сервису или группе сервисов. Если функция имя успешно выполняется, то назовем такой отказ восстанавливаемым 1-й очереди. Однако сервис или их группа, которым делегируются полномочия отказавшего сервиса, также могут не выполнить эту задачу, в случае чего оркестратор передает исполнение следующему подходящему сервису. Такой отказ будет называть 2-й и далее очереди.

Компенсация бизнес-транзакции – согласованное изменение состояний участников взаимодействия, приводящее бизнес-транзакцию к начальному состоянию. Скомпенсированным отказом назовем отказ, последствия которого были устранены таким образом, что атомарность бизнес-транзакции не была нарушена. Все невосстановленные и нескомпенсированные отказы являются авариями. Наработка бизнес-транзакции – количество успешных выполнений.

Для статистической оценки надежности предлагаем воспользоваться следующими показателями, которые представляют собой совмещенный набор показателей из традиционной теории надежности технических устройств, стандарта ISO 9126:2001, теории надежности систем «человек-техника» и впервые разработанные нами показатели с учетом свойств бизнес-транзакции и сервис-ориентированной среды (см. табл. 4). В этой таблице благодаря использованию понятия скомпенсированного и восстановленного отказа, мы можем также ввести статистически оцениваемый показатель атомарности.

Ввиду сложной природы бизнес-транзакции, по нашему мнению, использование показателей, получение которых связано с большим количеством тестовых испытаний, может быть затруднено. Это связано, например, с участием ручных операций, выполняемых оператором, или технической/организационной невозможностью тестирования на различных наборах данных.

Таблица 4 – Показатели атрибутов надежности внутренне-надежной бизнес-транзакции

Название	Описание	Формула	Описание формулы
Показатели стабильности			
Вероятность безаварийной работы	Вероятность того, что в пределах заданной наработки аварии не произойдет.	$X = \frac{A}{B}$	A – количество успешных выполнений; B – общее количество попыток выполнения.
Вероятность безотказной работы	Вероятность того, что в пределах заданной наработки отказа не произойдет.	$X = 1 - \frac{A}{B}$	A – количество отказов; B – общее количество попыток выполнения.
Частота отказов	Отношение числа возникших отказов к общему количеству попыток выполнения.	$X = \frac{A}{B}$	A – количество отказов; B – общее количество попыток выполнения.
Частота аварий	Отношение числа произошедших аварий к общему количеству попыток выполнения.	$X = \frac{A}{B}$	A – количество аварий; B – общее количество попыток выполнения.
Наработка на отказ	Количество успешных выполнений во возникновения первого отказа.	$X = \frac{A}{B}$	A – суммарное количество успешных выполнений; B – суммарное количество отказов.
Наработка на аварию	Количество успешных выполнений до возникновения первой аварии.	$X = \frac{A}{B}$	A – суммарное количество успешных выполнений; B – суммарное количество аварий.
Интенсивность отказов	Отношение количества произошедших отказов к общему количеству успешных выполнений.	$X = \frac{A}{B}$	A – количество отказов; B – количество успешных завершений.
Интенсивность аварий	Отношение количества произошедших аварий к общему количеству успешных выполнений.	$X = \frac{A}{B}$	A – количество аварий; B – количество успешных завершений.
Показатели отказоустойчивости			
Вероятность избежания аварий	Отношение количества произошедших аварий к числу отказов, их спровоцировавших.	$X = 1 - \frac{A}{B}$	A – количество аварий; B – количество отказов.
Вероятность устранения отказа	Вероятность того, что отказ может быть восстановлен.	$X = \frac{A}{B}$	A – количество восстановленных отказов; B – количество отказов.

Продолжение таблицы 4

Название	Описание	Формула	Описание формулы
Вероятность компенсации	Вероятность того, что невосстановленный отказ может быть успешно скомпенсирован.	$X = \frac{A}{B}$	A – количество успешных компенсаций отказа; B – количество невосстановленных отказов.
Показатели атомарности			
Частный уровень атомарности	Рассчитывается только для отдельной операции. Показывает вероятность того, что данная операция на нарушит согласованное состояние бизнес-транзакции.	$X = \frac{A - B + C}{A}$	A – общее количество попыток выполнения; B – количество отказов; C – количество восстановленных отказов.
Общий уровень атомарности	Вероятность того, что согласованное состояние системы не будет нарушено.	$X = \frac{A + B}{C}$	A – количество возвратов к начальному состоянию; B – количество успешных выполнений; C – общее количество попыток выполнения.
Показатели восстанавливаемости			
Разрешимость отказов	Вероятность того, что будет запущена попытка восстановления отказа.	$X = \frac{A_1}{A_2}$	A_1 – количество восстановленных отказов; A_2 – количество обнаруженных отказов.
Уровень восстанавливаемости	Вероятность успешного завершения восстановления.	$X = \frac{A}{B}$	A – количество успешных восстановлений; B – общее количество попыток восстановления.

Кроме того, часть показателей надежности связана с измерением времени работы программы или устройства. К анализу бизнес-транзакций такие показатели в большинстве не могут быть применены: время выполнения бизнес-транзакции может достигать больших величин, что соответствует ее свойствам, поэтому ее длительность не оказывает влияния на ее надежность. Исходя из этого, любые свойства надежности, связанные с оценкой времени не являются показательными для анализа надежности бизнес-транзакции.

Предлагаемые показатели предназначены для оценки надежности как бизнес-транзакции в целом, так и отдельных функций, участвующих в процессе ее выполнения. Однако это не исключает возможности и необходимости оценки надежности элементов, исполняющих эти функции. Особенности применения показателей для анализа надежности конкретной системы даны в 2.6.

Отметим важный факт, что для расчета большинства предлагаемых показателей используется либо показатель количества отказов, либо обратный к нему показатель количества успешных выполнений. То есть показатель количества отказов является важнейшим первичным показателем надежности.

Ввиду различной природы отказов бизнес-транзакции мы предполагаем, что каждый показатель может рассматриваться отдельно для каждого типа отказов, например, получением «аппаратной», «программной» или «организационной» восстанавливаемости. Заметим также, что надежность каналов связи рассматривается нами как аппаратная надежность средств обеспечения сервисов ввиду того, что каждое выполнение бизнес-транзакции в сервис-ориентированной среде, особенно выполнение с отказами, влечет за собой появление сложнопредсказуемых временных связей между службами – это последствия слабой связанности и независимости служб. Поэтому отказ канала связи между парой служб или оркестратором и службой отнесем к отказу оборудования.

Обратим внимание на расчет показателей надежности при различных способах объединения участников при выполнении задачи. Проблемы ветвления потока управления бизнес-процессов рассмотрены в [193], [211], а в [212] приводится расчет стабильности интегрированной системы, включающей множество функционально дублирующих друг друга элементов (резервирование). Кроме этого, как уже отмечалось, традиционная теория надежности рассматривает способы расчета надежности структурно-сложных систем, включающих как параллельное и последовательное соединение, так и нетривиальные схемы с различными методиками расчета стабильности таких структур, схем восстановления и резервирования ([213, 214, 164, 162, 215] и др.). Ввиду практически неограниченных вариантов объединения сервисов и высо-

кой динамики этой структуры применение этих методов сопряжено с большими временными и трудовыми затратами.

Следует учитывать, что сервис-ориентированная среда позволяет заменять сбойную операцию или сервис другим участником или группой операций, выполняющих аналогичную задачу и возвращающих аналогичный результат, поэтому попытка восстановления связана не только с повторным обращением или перезапуском операции, но и, например, с обращением к схожему по функциональности участнику, который изначально не принимал в транзакции участия. При условии некоторого постоянства количества и качества отказов для распределенных процессно-ориентированных систем параметрами, на который оказывает существенное влияние сервис-ориентированная архитектура, являются отказоустойчивость и восстанавливаемость. Стабильность для отдельных элементов неотличима от аналогичных показателей программных и технических средств, однако в совокупности свойства СОА позволяют повысить именно отказоустойчивость и восстанавливаемость системы в целом, и, естественно, повлиять таким образом на ее стабильность. И, исходя из того, что отказ участника или операции транзакции не входит в область ее ответственности, то атрибутами внутренне-надежной транзакции можно признать только атомарность, отказоустойчивость и восстанавливаемость.

В сервис-ориентированной бизнес-среде существует возможность повышения надежности путем улучшения способности бизнес-транзакции к восстанавливаемости через возможность «горячей» замены сбойного участка процесса на другой, доступный и аналогичный по функциональности. В СОА это можно реализовать путем вызова функций сервиса или группы сервисов, реализующих те же операции, что служба, в доступе к которой возникли проблемы. При этом необходимо учитывать природу сервиса как бизнес-процесса, а именно невозможность отмены некоторых операций, длительность операции и низкое время реакции, высокую вероятность отказа в выполнении и др.

2.3. Окрашенные сети Петри

Исследование бизнес-транзакций возможно с точки зрения различных аспектов, для чего могут применяться различные средства моделирования. К таким аспектам изучения, как уже было показано, относятся веб-сервисы, сервис-ориентированная архитектура, транзакционное управление, композиции сервисов, поиск сервисов в распределенной среде, различные технологии интеграции, системы параллельных процессов, и, наконец, бизнес-процессы.

Однако для рассмотрения вопросов моделирования надежной бизнес-транзакции используемые техники моделирования не позволяют описывать бизнес-процесс вместе с атрибутами, присущими надежности: атомарностью, отказоустойчивостью, восстанавливаемостью, стабильностью.

Поэтому необходимо разработать аппарат моделирования, позволяющий:

- Представлять транзакцию как бизнес-процесс и как композицию разнородных сервисов одновременно.
- Исследовать проблемы поддержки атомарности транзакции за счет отмены изменений и/или их компенсации.
- Исследовать уровень отказоустойчивости и восстанавливаемости за счет возможности динамической реконфигурации транзакции, повторного вызова сбойных сервисов, частичной компенсации и других путей выполнения.
- Исследовать показатели стабильности транзакции путем внедрения стохастических и временных параметров модели, позволяющих оценивать поведение транзакции в реальной ситуации.

Для построения аппарата моделирования в качестве базового инструмента воспользуемся теорией окрашенных сетей Петри. Для этого дадим краткое описание теории окрашенных сетей Петри.

Окрашенной сетью Петри [82] будем называть девятку

$$CPN = (P, T, A, \Sigma, V, C, G, E, I), \quad (1)$$

где P – конечное множество позиций;

T – конечное множество переходов таких, что $P \cap T = \emptyset$;

A – конечное множество направленных дуг, таких, что $P \cap T = P \cap A = T \cap A = \emptyset$;

Σ – конечное множество непустых типов, также называемых множествами цветов. Множество типов определяет значения данных, функций и операторов (выражения дуг, стражей и функций инициализации);

V – конечное множество типизированных переменных таких, что $\forall v \in V: Type[v] \in \Sigma$. Под обозначением $Type$ понимается функция, возвращающая тип переменной;

$C: P \to \Sigma$ – функция «окрашивания», ставящая в соответствие каждой позиции подмножество цветов, т.е. интуитивно это означает, что каждая фишка позиции p должна иметь типы данных, принадлежащие $C(p)$;

G – сторожевая функция, ставящая в соответствие каждому переходу сторожевое условие («стража»), такая, что: $Type[G(t)] = Bool$, то есть функция возвра-

щает булевое значение, а все переменные выражения имеют тип, принадлежащий Σ;

E – функция выражения над дугой, ставящая в соответствие каждой дуге $a \in A$ выражение такое, что $\forall a \in A : Type[E(a)] = C(p)_{ms}$, где p – это позиция, соединенная с дугой a; Это означает, что каждое выражение над дугой должно возвращать комплект над типом смежной позиции.

I – это функция инициализации, ставящая в соответствие каждой позиции $p \in P$ выражение инициализации такое, что $\forall p \in P: Type[I(p)] = C(p)_{ms}$.

Окрашенная сеть Петри напрямую не разрешает выделение параллельных дуг, заменяя отображаемый ими функционал комплектами переменных и операциями над дугами, т.е. граф окрашенной сети Петри представляет собой простой граф.

Для окрашенной сети Петри $CPN = (P, T, A, \Sigma, V, C, G, E, I)$ определяются также следующие концепции [82]:

- Метка (фишка) – это пара (p, c), где $p \in P$ и $c \in C(p)$. Множество всех меток обозначим как TE;
- Маркировка – это функция M, которая отображает каждую позицию $p \in P$ в комплект меток $M(p) \in C(p)_{ms}$;
- Начальная маркировка M_0 определяется через вычисление инициализирующих выражений, т.е. $M_0(p) = I(p)\langle\,\rangle$ для всех $p \in P$;
- Переменные перехода t обозначаются как $Var(t) \subseteq V$ и содержат свободные переменные, используемые в страже перехода t и выражениях над дугами, соединенными с t.
- Привязка перехода t – это функция b, которая отображает каждую переменную $v \in Var(t)$ в значение $b(v) \in Type(v)$. Множество всех привязок перехода t обозначим как $B(t)$.
- Элемент привязки – это пара (t, b) такая, что $t \in T$ и $b \in B(t)$. Множество всех элементов привязки $BE(t)$ для перехода t определяется как $BE(t) = \{(t, b) | b \in B(t)\}$. Через BE будем обозначать множество всех элементов привязки CPN.
- Шаг $Y \in BE_{ms}$– это непустой, конечный комплект элементов привязки.

Обозначим через $G(t)\langle b\rangle$ результат вычисления выражения сторожевой функции $G(t)$ перехода t привязки b. Аналогично обозначим за $E(a)\langle b\rangle$ результат вычисления выражения над дугой $E(a)$ дуги a привязки b.

Элемент привязки $(t, b) \in BE$ разрешен в маркировке M тогда и только тогда, когда выполняются следующие условия:

– $G(t)\langle b\rangle = true;$ (2)

– $\forall p \in P: E(p,t)\langle b\rangle \ll= M(p).$

Когда элемент (t, b) разрешен в M, он может быть запущен, что приведет к маркировке M', определяемой как:

– $\forall p \in P: M'(p) = (M(p) - -E(p,t)\langle b\rangle) + +E(t,p)\langle b\rangle.$ (3)

Шаг Y включает все элементы привязки для одного перехода. Все элементы привязки шага должны быть способны удалить все метки из соответствующей позиции. Таким образом, каждая позиция p должна быть отмечена комплектом меток $M(p)$, который больше или равен сумме всех меток, которые удаляют из p отдельными элементами привязки шага Y. Т.е для всего шага Y:

$$\overset{++}{_{MS}}\sum_{(t,b)\in Y} E(p,t)\langle b\rangle \ll= M(p) \quad (4)$$

Символ MS показывает комплект комплектов. Каждое выражение $E(a)\langle b\rangle$ происходит столько же раз, сколько запускается элемент (t, b) в шаге Y. Поэтому можно сказать, что шаг $Y \in BE_{ms}$ разрешен в маркировке M тогда и только тогда, когда выполняются следующие условия:

– $\forall (t,b) \in Y: G(t)\langle b\rangle = true.$

– $\forall p \in P: \overset{++}{_{MS}}\sum_{(t,b)\in Y} E(p,t)\langle b\rangle \ll= M(p).$

Когда шаг Y разрешен, он может быть запущен, что приведет к маркировке M', определяемой как:

– $\forall p \in P: M'(p) = (M(p) - - \overset{++}{_{MS}}\sum_{(t,b)\in Y} E(p,t)\langle b\rangle) \; + + \overset{++}{_{MS}}\sum_{(t,b)\in Y} E(t,p)\langle b\rangle$ (5)

Если маркировка M_2 была достигнута после срабатывания шага Y из маркировки M_1, то говорят, что M_2 напрямую достижима из M_1: $M_1 \overset{Y}{\rightarrow} M_2$ или $M_1 \longrightarrow M_2$. При этом возможно построение конечной последовательности длинной n вида:

$$M_1 \overset{Y_1}{\rightarrow} M_2 \overset{Y_2}{\rightarrow} M_3 \ldots M_n \overset{Y_n}{\rightarrow} M_{n+1} \quad (6)$$

такой что $M_i \overset{Y_i}{\rightarrow} M_{i+1}$ для всех $1 \le i \le n$. Обозначим конечную последовательность шагов Y из маркировки M_1 к M_n как $M_1 \overset{Y}{\Rightarrow} M_n$. Любая из представленных маркировок достижима из M_1. Набор достижимых маркировок из M обозначим за $\mathcal{R}(M)$.

Будем использовать запись $M_1 \overset{Y}{\rightarrow}$ для обозначения, что шаг Y из M_1 разрешен, но маркировка, к которой он приводит, не имеет для нас значения в данный момент.

Домашняя маркировка окрашенной сети Петри M_{home} – это маркировка, которая может быть достигнута из любой достижимой маркировки, т.е. $\forall M \in \mathcal{R}(M_0): M_{home} \in \mathcal{R}(M)$.

Пространство домашних маркировок задает множество маркировок, которые достижимы из любой достижимой, т.е. M^*_{home} – это домашнее пространство, если $\forall M \in \mathcal{R}(M_0)\ \exists M' \in \mathcal{R}(M): M' \in M^*_{home}$.

Мертвая маркировка – это маркировка без активных переходов из нее, т.е. $\forall t \in T: \neg(M_1 \overset{t}{\rightarrow})$. Переход называется мертвым для маркировки M_0 тогда и только тогда, когда $\forall M \in \mathcal{R}(M_0): \neg(M_1 \overset{t}{\rightarrow})$. Переход называется живым для маркировки M_0 тогда и только тогда, когда $\forall M \in \mathcal{R}(M_0) \exists M' \in \mathcal{R}(M): M' \overset{t}{\rightarrow}$. Мертвая маркировка также может называться пассивной.

Позицию $p \in P$ будем называть k -ограниченной, если $\forall M \in \mathcal{R}(M_0): |M(p)| \leq k$. Сеть Петри является k -ограниченной, если каждая ее позиция k-ограниченна. Если нас не интересует конкретное значение границы, то сеть Петри или позиция сети является ограниченной, если существует некоторое счетное k для которого сеть или позиция является k-ограниченной. Комплект m будем называть граничным комплектом позиции $p \in P$, если $\forall M \in \mathcal{R}(M_0): |M(p)| \leq m$. По аналогии комплект m может быть граничным комплектом всей сети.

Преимущества использования окрашенных сетей Петри для моделирования динамических систем были описаны Куртом Йенсеном (Kurt Jensen) в его тематических работах (например, в [70], [82], [216]).

2.4. Аппарат моделирования бизнес-транзакции

Представим разработанный нами аппарат для моделирования бизнес-транзакций, который основывается на теории окрашенных сетей Петри. Аппарат определяет основные термины, которые используются для построения бизнес-транзакции, описания ее структуры, связей, элементов и среды функционирования.

2.4.1. Выделение сервиса и сервисной операции

Как было показано нами в [217], сети Петри используются для моделирования сервисных композиций и бизнес-процессов. Идея бизнес-транзакций в сервис-ориентированной среде позволяет представить переходы за отдельные сервисы или вложенные бизнес-процессы, тогда как маркировки позиции демонстрируют определенные состояния, достижимые бизнес-транзакцией в процессе выполнения.

Ввиду того, что сервис как программная единица может реализовывать множество различных операций, целесообразно рассматривать в качестве переходов сети

Петри эти операции. В сервис-ориентированном программировании существует ряд способов выделения и группировки этих операций. Например, в стандарте WSDL, операции объединяются в некоторые наборы (т.н. «типы порта»), которые уже включаются в сервис [218]. Операция сервиса реализует отдельную задачу с четко определенными входными и выходными параметрами. Таким образом, каждую операцию можно представить в виде отдельного перехода и учитывать то, что в модели, реализованной с помощью сети Петри, несколько переходов могут принадлежать одному и тому же сервису. Как уже отмечалось, контракт службы является универсальной технологией обеспечения слабой связанности, отделяя ее описание выполняемых ею функций от конкретной реализации.

2.4.2. Операция службы и участник бизнес-транзакции

Определение 1. Пусть $CPN = (P, T, A, \Sigma, V, C, G, E, I)$ – это окрашенная сеть Петри, моделирующая взаимодействие сервисов [219]. Тогда отдельную *операцию службы* мы представим как:

$$O = (t, P^o, A^o, \Sigma^o, V^o, C^o, G^o, E^o), \quad (7)$$

где $t \in T$ – отдельный переход сети Петри;

$P^o \subseteq P$ – множество входных и выходных позиций такое, что $P^o = P_{in} \cup P_{out}$;

$P_{in} \subseteq P^o$– множество входных позиций перехода t;

$P_{out} \subseteq P^o$множество выходных позиций перехода t;

$A^o \subseteq P_{in} \times t \cup t \times P_{out}$ – множество дуг, соединяющих переход t с входными и выходными позициями. Под A_{in} и A_{out} будем понимать набор входящих и исходящих дуг соответственно;

$\Sigma^o \subseteq \Sigma$ – это подмножество множества типов данных, используемых в окрашенной сети Петри;

$V^o \subseteq V$ – подмножество типизированных переменных, используемых операцией;

$C^o: P^o \to \Sigma^o$ – функция «окрашивания», ставящая в соответствие каждой позиции подмножество цветов;

G^o – сторожевая функция, ставящая в соответствие операции сторожевое условие, такая, что: $Type[G^o(t)] = Bool$, то есть функция возвращает булевое значение;

E^o – функция выражения над дугой, ставящая в соответствие каждой дуге $a \in A^o$ выражение такое, что $\forall a \in A^o : Type[E^o(a)] = C(p)_{ms}$, где p – это позиция, со-

единенная с дугой a. Под E_{in} и E_{out} будем понимать множество выражений над входящими и исходящими дугами соответственно.

Единственное сторожевое условие позволяет осуществлять запуск операции в случае, если входные переменные будут удовлетворять условиям, задаваемым стражами. Функция окрашивания $C: P^o \to \Sigma$ позволяет задавать строго типизированный вход и выход операции, что присуще свойствами контракта службы и задачам, которые она реализует.

Операция O является ограниченной, если для всех ее входных позиций определен некоторый граничный комплект: $\forall p \in P^o_{in}, \forall M\ (p): |M(p)| \ll= m_p$, где m_p – граничный комплект позиции p. Требования ограниченности операции необходимо, например, если отдельная операция представляется системой массового обслуживания с накопителем ограниченной емкости. Сеть является ограниченной, если все ее операции ограниченны.

Сервис, как и любая программная операция, может вызываться только одним участником. Если сервис вызывается несколькими участниками, то происходит запуск нескольких служб одновременно. Таким образом, для сети Петри необходимо задать ограничение, предполагающее вызов сервиса только одним участником. Для этого необходимо задать условие, чтобы все входные позиции операции принадлежали одному участнику.

Введем понятие участника. Пусть H – множество участников сервисного взаимодействия. Множество позиций и операций принадлежат этим участникам. Принадлежность будем обозначать как ${}_hP$ – где $h \in H$, что означает множество операций P принадлежит участнику h (вернее операции входят в состав сервисов, которые принадлежат участнику). Пусть $CPN = (P, T, A, \Sigma, V, C, G, E, I)$ – это окрашенная сеть Петри, моделирующая взаимодействие сервисов, H – множество участников. Тогда:

$$- \quad P = {}_{h_1}P \cup {}_{h_2}P \cup \ldots \cup {}_{h_n}P, \tag{8}$$

$$- \quad T = {}_{h_1}T \cup {}_{h_2}T \cup \ldots \cup {}_{h_n}T,$$

где n – количество участников, P и T – множество переходов и операций сети соответственно.

Пусть $O = (t, P^o, A^o, \Sigma^o, V^o, C^o, G^o, E^o)$ – это отдельная операция сервиса сети CPN, H – множество участников. Чтобы удовлетворять природе вызываемых программных операций, эта операция должна быть ограничена следующим условием:

$$- \quad P^o_{in} \subseteq {}_hP: h \in H, \tag{9}$$

то есть все входные позиции операции должны принадлежать одному участнику.

Предусловия операции (т.е. условия, которые должны быть выполнены перед запуском операции) выражаются с помощью функции окрашивания, задающей определенные типы данных входной позиции и выражений над входными дугами, определяющими наборы входящих данных и/или их преобразования. Т.е. предусловие можно представить как $R_{in} = (P_{in}, A_{in}, E_{in}, \Sigma'_{in}, V'_{in}, C_{in})$. Аналогично *представляются постусловия операции* (т.е. условия, которые должны быть выполнены при завершении операции): $R_{out} = (P_{out}, A_{out}, E_{out}, \Sigma'_{out}, V'_{out}, C_{out})$. Таким образом, операцию службы можно выразить в терминах пост- и предусловий: $O = (t, R_{in}, R_{out}, G)$.

Ввиду того аппарат окрашенных сетей Петри не позволяет описывать вычисления над данными вне операций над дугами, то выполняемые операцией функции можно представить только в виде совокупности операций над входными и выходными дугами.

2.4.3. Интерфейс и композиция операций

Исходя из определения сети Петри, одна сервисная операция с входами и выходами может представлять собой сеть Петри, являясь простым двудольным ориентированным графом.

Пусть $CPN = (P, T, A, \Sigma, V, C, G, E, I)$ – это окрашенная сеть Петри, моделирующая взаимодействие сервисов, а $O = (t, P^o, A^o, \Sigma^o, V^o, C^o, G^o, E^o)$ – это отдельная операция сервиса.

При этом операции могут разделять общие позиции. Позиция p операций O и O' является общей между ними в том случае, когда $p \in (P^o \cap P^{o'})$.

Определение 2. Под *интерфейсом операции* будем понимать совокупность ее входных и выходных позиций, т.е. P^o. Назовем две операции O и O' *интерфейсно-эквивалентными* в том случае, когда: $C(\forall p^o \in P^o) = C(\forall p^{o'} \in P^{o'})$. Для обозначения эквивалентности по интерфейсу будем использовать выражение $O\ eq\ O'$.

Две операции O и O' назовем *интерфейсно-совместимыми* в том случае, когда выходные позиции операции O могут являться входными для операции O', т.е.:
$$(\forall p \in P^o_{in} \wedge \forall p' \in P^{o'}_{out}): C(p) = C(p') \quad (10)$$

Две операции O и O' назовем частично интерфейсно-совместимыми в том случае, когда подмножество выходных позиций операции может входить в подмножество выходных для другой операции, т.е.:
$$P^o_{in} \cap P^{o'}_{out} \neq \emptyset \quad (11)$$

Операции O и O'назовем *несовместимыми*, если $P^o_{in} \cap P^{o'}_{out} = \emptyset$

Необходимо учитывать, что интерфейсная совместимость операций не гарантирует, вообще говоря, их функциональной совместимости. Т.е. операции могут быть совместимы по входным и выходным данным, но выполнять различные несовместимые функции.

Определение 3. *Композицией операций* O и O' назовем окрашенную сеть Петри

$$O \oplus O' = (T, P, A, \Sigma, V, C, G, E), \quad (12)$$

где $T = (t^o, t^{o\prime})$;

$P = P^o \cup P^{o\prime}$, причем $P = P^{\mathrm{o}}_{in} \cup P^{o'}_{out} \cup P^{o'}_{in} \cup P^{\mathrm{o}}_{out}$, $P_{in} = (P^{\mathrm{o}}_{in} \backslash P^{o'}_{out}) \cup (P^{o'}_{in} \backslash P^{\mathrm{o}}_{out})$, $P_{out} = (P^{\mathrm{o}}_{out} \backslash P^{o'}_{in}) \cup (P^{o'}_{out} \backslash P^{\mathrm{o}}_{in})$. Для интерфейсно-совместимых операций в этом случае $P^{\mathrm{o}}_{in} = P^{o'}_{out}$: $P_{in} = P^{\mathrm{o}}_{in}$, $P_{out} = P^{o\prime}_{out}$;

$A = A^o \cup A^{o\prime}$;

$\Sigma = \Sigma^o \cup \Sigma^{o\prime}$;

$C = \{ C^o, C^{o\prime}\}$;

$G = \{ G^o, G^{o\prime}\}$;

$E = \{ E^o, E^{o\prime}\}$.

Композицией конечного множества частично интерфейсно-совместимых операций $O_1 \dots O_n$ и O' для всех $n \geq 1$ назовем окрашенную сеть Петри $\{O_1 \dots O_n\} \oplus O' = (T, P, A, \Sigma, V, C, G, E)$, (13)

где $T = (t^{O1} \dots t^{On}, t^{O\prime})$;

$P = P^{o1} \cup \dots \cup P^{on} \cup P^{o\prime}$, причем $P_{in} = (P^{O\prime}_{in} \backslash (P^{\mathrm{O1}}_{out} \cup \dots \cup P^{\mathrm{On}}_{out})) \cup ((P^{\mathrm{O1}}_{in} \cup \dots \cup P^{\mathrm{On}}_{in}) \backslash P^{O\prime}_{out})$, $P_{out} = (P^{O\prime}_{out} \backslash (P^{\mathrm{O1}}_{in} \cup \dots \cup P^{\mathrm{On}}_{in})) \cup ((P^{\mathrm{O1}}_{out} \cup \dots \cup P^{\mathrm{On}}_{out}) \backslash P^{O\prime}_{in})$;

и далее по аналогии с (4).

Будем говорить для (5), что имеет место полная совместимость для конечного множества частично интерфейсно-совместимых операций $O_1 \dots O_n$ и O' по входу, если: $P^{O\prime}_{in} \not\subseteq P_{in}$, и по выходу, если $P^{O\prime}_{out} \not\subseteq P_{out}$.

2.4.4. Условия существования декомпозирующих сетей

Сервис можно представить как множество операций O, объединенных общностью бизнес-логики программы. Сервис может реализовывать совершенно различные функции, поэтому к множеству операций невозможно применить требования общности пост- или предусловий. При этом разные операции одного сервиса могут входить в разные сервисные композиции (моделируемые разными сетями Петри). Верно и другое: разные операции сервиса могут входить в одну модель.

Исходя из природы сервиса и теории оркестровки, каждая операция может быть рекурсивно декомпозирована на ряд вложенных функций до достижения уровня элементарных операций. Рассмотрим операцию O как вложенную окрашенную сеть Петри CPN', содержащую непустое множество переходов. Все предусловия операции O необходимо представить элементами, задающими начальную маркировку CPN', а постусловия задают финальную маркировку.

Определение 4. Пусть $CPN = (P, T, A, \Sigma, V, C, G, E, I)$ – это окрашенная сеть Петри, моделирующая взаимодействие сервисов, а $O = (t, P^o, A^o, \Sigma^o, V^o, C^o, G^o, E^o)$ – это отдельная операция сервиса. Тогда для O будем называть сеть $CPN' = (P_s, T_s, A_s, \Sigma_s, V_s, C_s, G_s, E_s, I_s)$ *декомпозирующей сетью* 1-го порядка в случае, если: (7)

1. $\forall p_{in} \in P_{in} \exists M_0(p_s): M_0(p_s) = M(p_{in})$ и $C(p_{in}) = C(p_{s_{in}})$, где $P_{in} \subseteq P^o$, $p_s \in P_s$, а для всех остальных $p_s \in P_s$ выполняется условие $M_0(p_s) = \emptyset$.
2. $\forall p_{out} \in P_{out} \exists M_f(p_s): M_f(p_s) = M(p_{out})$ и $C(p_{out}) = C(p_{s_{out}})$, где $p_s \in P_s$, под M_f понимается финальная маркировка, такая что:
3. Существует конечная последовательность шагов, таких что $M_0(p_s) \longrightarrow M_f(p_s)$, т.е. финальная маркировка достижима из начальной маркировки $M_f \in \mathcal{R}(M_0)$ и $R(M_f) = \emptyset$, т.е. финальная маркировка мертва.

Таким образом, все входные позиции операции отображаются как начальная маркировка для вложенной сети, при этом остальные позиции не имеют маркировки. Все выходные позиции формируются из финальной маркировки вложенной сети, достижимой из начальной.

Будем называть сеть CPN^k декомпозирующей сетью k-го порядка в случае, если она декомпозирует операцию O, входящую в сеть $(k-1)$-го порядка.

Будем называть операцию *атомарной* в случае, если она реализуется одним переходом, и *композитной*, если она реализуется декомпозирующей сетью.

Отметим, что сеть Петри, реализующая операцию, зависит от внешней среды только через интерфейс операции P^o, что позволяет реализовать концепцию слабой связанности.

Для ограничения операции в рамках бизнес-процесса воспользуемся условиями, налагаемыми на сеть Петри WF-сетями. Первое условие WF-сети противоречит возможности операции иметь множество входных и выходных позиций, поэтому оно не может быть применено. Однако второе условие позволяет наложить на операцию, представленную окрашенной сетью, условие отсутствия бесконечных циклов и сег-

ментов сети, не связанных с достижением целевой маркировки. Поэтому наложим на сеть Петри, реализующую операцию, 4-е условие:

4. $\forall t \in T_S: \exists Y(t): M' \xrightarrow{Y} \ldots \rightarrow M_f$, где $(M', M_f) \in \mathcal{R}(M_0)$. (14)

Отметим, что определение операции в виде низкоуровневой сети Петри, позволяет реализовать рекурсивную декомпозицию этой сети на операции (сети) более низкого уровня. Верно и обратное: сеть Петри, удовлетворяющей условиям сервисной операции, можно представить, как реализацию некоторой высокоуровневой функции.

2.4.5. Особенности разработанного аппарата моделирования

Методика выделения из службы набора операций позволяет реализовать модели сервисного взаимодействия на сетях Петри без ограничений, связанных со сложностью службы.

В отличие от представленных в западной литературе подходов в виде: «сервисных» или «открытых» сетей Петера Массутхе (например, в [75]), WF-сетей Вилла ван дер Аалста [73] и простых сетей Петри, получаемая модель обладает следующими преимуществами:

- Предлагает рассматривать сервис и с точки зрения программной реализации, то есть позволяет описывать не сервисы целиком, а отдельные сервисные операции, которые, возможно, не связаны друг с другом.
- Выделяет ключевые аспекты сервис-ориентированного программирования: понятие интерфейса, независимой реализации службы, сервисной композиции, которая может рассматриваться как единый сервис с более высокого уровня абстракции и также включать в композицию с другими службами.
- Описывается на базе высокоуровневых сетей Петри – окрашенных сетей, что позволяет использовать все преимущества этой технологии для моделирования динамических систем, в первую очередь, получаемая модель, в отличие от простых сетей Петри, способна описывать манипуляции с различными типами данных.
- Не имеет ограничений, накладываемых на процесс выполнения на начальные и финальные позиции модели, что позволяет считать взаимодействие завершенным при достижении финальной маркировки, а не некоторой финальной позиции. Верно и обратное – инициировать взаимодействие позволяет

начальная маркировка, охватывающая, возможно, множество сервисов, а не некоторая начальная позиция.

- С другой стороны, модель позволяет отображать не только исключительно сервис-ориентированные системы, но, что важно, описывать любые многоэтапные бизнес-процессы, которые включают ряд последовательных операций.

2.5. Описание бизнес-транзакции на языке окрашенных сетей Петри

2.5.1. Формализованное описание бизнес-транзакции

Определение сервиса и сервисной операции с помощью аппарата неиерархических окрашенных сетей Петри позволяет нам выразить бизнес-транзакцию теоретико-множественной формулой [146, 220, 221]:

Определение 5. Сеть Петри $BT = (P, T, A, \Sigma, V, C, G, E, I, H, \Omega)$ можно представить как *бизнес-транзакцию* в сервис-ориентированной среде, если: (15)

- P – конечное множество позиций бизнес-транзакции, среди которых выделяются промежуточные P_m и стабильные позиции P_f такие, что $P = P_m \cup P_f$; Стабильные позиции можно разделить на входные P_{in} и выходные P_{out}, где $\bullet P_{in} \cap P_{out} \bullet = \emptyset$.
- T – конечное множество операций, при этом каждая операция может быть представлена как подсеть Петри или декомпозирована до самостоятельной сети;
- A – конечное множество направленных дуг, таких, что $P \cap T = P \cap A = T \cap A = \emptyset$;
- Σ – конечное множество непустых типов данных, используемых в транзакции.
- V – конечное множество типизированных переменных таких, что $\forall v \in V: Type[v] \in \Sigma$;
- $C: P \to \Sigma$ – функция «окрашивания», ставящая в соответствие каждой позиции подмножество типов данных.
- G – сторожевая функция, ставящая в соответствие каждому переходу сторожевое условие.
- E – функция выражения над дугой, ставящая в соответствие каждой дуге $a \in A$ опередеенное выражение.

- I – это функция инициализации, ставящая в соответствие каждой позиции $p \in P$ выражение инициализации такое, что $\forall p \in P: Type[I(p)] = C(p)_{ms}$.
- Пара (p, c), где $p \in P$ и $c \in C(p)$ – это метка (фишка) сети, определяющая нахождение данных определенного типа в определенной позиции.
- Маркировка – это комплект меток в позициях сети. Маркировка сети определяет состояние бизнес-транзакции.
- Начальная маркировка M_0 определяется через вычисление инициализирующих выражений, т.е. $M_0(p) = I(p)$ для всех $p \in P$ и является начальным состоянием всех участников бизнес-транзакции.
- Финальная маркировка M_f достигается через конечную последовательность шагов Y, при этом $M_f \in \mathcal{R}(M_0)$ и $\mathcal{R}(M_f) = \emptyset$. Множество финальных маркировок обозначим за Ω.
- H – конечное множество участников бизнес-транзакции.

Остальные значения бизнес-транзакции аналогично объясняются в терминах окрашенной сети Петри. Таким образом, выполнение бизнес-транзакции заключается в последовательном переходе от начальной к финальной маркировке, последующие переходы из которой невозможны. Каждый переход является атомарной операцией, которая, как уже отмечалось, может быть рассмотрена в виде сети Петри, на которую налагается ряд ограничений.

С другой стороны, если $BT = (P, T, A, \Sigma, V, C, G, E, I)$ выполняет требования сервисных операций, то она может рассматриваться как сервисная операция и включаться в качестве атомарной функции в состав композиций более высокого уровня.

Как можно увидеть, окрашенная сеть Петри детерминировано определяет бизнес-транзакцию и ее поведение. Фактор случайности присутствует только при выборе очередности из списка равновероятных активных переходов сети. В целях имитационного моделирования необходимо анализировать работу транзакции вместе с отказами операций. Поэтому во время имитации специально разработанным программным обеспечением (см. главу 2.9) в модель бизнес-транзакции вносятся специальные изменения, которые позволяют инициировать отказы ее отдельных операций с заданной вероятностью.

2.5.2. Атомарная, восстанавливаемая и стабильная бизнес-транзакция

Пусть $BT = (P, T, A, \Sigma, V, C, G, E, I, H, \Omega)$ – это бизнес-транзакция в сервис-ориентированной среде $\mathcal{E}$, в которой также существует множество сервисов S, каж-

дый из которых, в свою очередь, реализует конечное множество атомарных операций O. Обозначим множество операций сервиса $S_1 = O^{s1}$, $S_2 = O^{s2}\ldots$, $S_n = O^{sn}$, при этом $O = O^{s1} \cup O^{s2} \cup \ldots \cup O^{sn}$, где $n \geq 1$ показывает количество служб в среде бизнес-транзакции. Интерфейсно-совместимые службы могут составлять конечное множество композиций Q.

Тогда (как нами показано в [222]):

- Ввиду определения для BT выполняется условие $T \subseteq O$.
- Определение 6. Будем называть операцию O' *восстанавливаемой* 1-го порядка, если существует композиция q атомарных операций, такая, что она интерфейсно-эквивалентна O': $\exists q = O_1 \oplus O_2 \ldots \oplus O_n,\ \forall 1 \leq i \leq n, O_i \in O: q\ eq\ O'$. Иными словами для O' существует декомпозирующая сеть 1-го порядка, построенная на основе операций $O_1 \ldots O_n$. Такую композицию будет называть восстанавливающей композицией.

  Иначе говоря:

  Пусть $\exists M', M'' \in \mathcal{R}(M_0): M''(p) = \left(M'(p) - -E_{in}^{O'}\right) + +E_{out}^{O'}$ для $\forall p \in P^O$, тогда если $\exists q = O_1 \oplus O_2 \ldots \oplus O_n,\ \forall 1 \leq i \leq n, O_i \in O:\ M''(p) = \left(M'(p) - -E_{in}^{O_1} - -E_{in}^{O_2} - -.. - - E_{in}^{O_n}\right) + +E_{out}^{O_1} + +E_{out}^{O_2} + +.. + + E_{out}^{O_n}$, то q – восстанавливающая композиция. (16)
- Будем называть операцию O' восстанавливаемой k -го порядка, если существует композиция атомарных и/или композитных операций такая, что она интерфейсно-эквивалентна O', при этом композитные операции могут состоять из декомпозирующих сетей вплоть до k-го порядка.
- Будем называть бизнес-транзакцию BT *восстанавливаемой* k-го порядка, если она состоит только из восстанавливаемых операций k -го порядка.
- Будем называть бизнес-транзакцию BT частично восстанавливаемой k-го порядка, если она включает восстанавливаемые операции вплоть до k -го порядка.
- Исходя из того, что бизнес-транзакция представляет собой композицию взаимодействующих объектов реального мира, то ясно, что каждый объект не должен обрабатывать неограниченное число входящих ресурсов, иначе говоря, каждая операция и бизнес-транзакция в целом должны быть ограниченны. Это свойство присуще любой бизнес-транзакции, не влияет на ее надежность, но должно соблюдаться при ее моделировании.

- Бизнес-транзакция должна обладать полностью предсказуемым поведением, несмотря на стохастичность поведения ее участников. Т.е. множество домашних маркировок бизнес-транзакции должно быть конечное и, строго говоря, все его элементы должны быть известны проектировщику. А в идеале бизнес-транзакция должна быть вообще атомарной.
- Определение 7. Будем называть бизнес-транзакцию BT *атомарной* тогда, когда ее начальная M_0 и финальные маркировки M_f являются домашними маркировками, т.е. $M_0 \in M^*_{home}$ и $M_f \in M^*_{home}$. Таким образом, атомарность транзакции предполагает, что ее маркировки, не являющиеся домашними или финальными, не мертвы.
- Если для некоторого состояния M' бизнес-транзакции BT существует композиция q операций с такими маркировками M^q, что $M' = M_0^q$ и $M_f^q = M_0$, т.е.
  $$\exists M' \in \mathcal{R}(M_0), \exists q \in Q: M' = M_0^q \, , M_f^q = M_0 \quad (17)$$
  то назовем состояние M' компенсируемым состоянием, композицию q – компенсирующей сетью или композицией и будем обозначать переход между маркировками с помощью q как $M' \xrightarrow{q} M''$.
- По аналогии с восстанавливаемостью, частичной композиционной атомарностью транзакция может обладать, если для некоторых ее маркировок существуют компенсирующие композиции.
- Будем называть бизнес-транзакцию компенсируемой k –го порядка, если все ее маркировки, кроме начальной и финальной – компенсируемые сетями вплоть до k-го порядка. Понятно, что компенсируемая бизнес-транзакция является атомарной. Но не наоборот.

Пусть $BT = (P, T, A, \Sigma, V, C, G, E, I, H, \Omega)$ – это бизнес-транзакция, O' – некоторая реализуемая ею операция с комплектами меток на входных и выходных позициях $M(P_{in}^O)$ и $M(P_{out}^O)$, O – декомпозирующая сеть операции O', и существует компенсирующая сеть q такая, что $M' \xrightarrow{q} M''$. Пусть $M' = M\left(P_{in}^O\right) + + M(P_{other})$ и $M'' = M(P_{out}^O) + + M(P_{other})$, где P_{other} – другие позиции сети, не относящиеся к O', такие что $P = P_{in}^O \cup P_{out}^O \cup P_{other}$. Тогда очевидно, что O', O, q эквивалентны по своим интерфейсам, то есть компенсирующая сеть, изменяющая позиции одного перехода является декомпозирующей сетью для него. На практике это означает, что компенсирующие сети могут восстанавливать операции, и не всегда компенсация является «обратной» процедурой к некоторой операции.

Отметим, что условие (4) сервисной композиции не ограничивает возможность появления в ней блокировок и циклов. Путь от начальной до финальной маркировки может существовать, однако такая модель не гарантирует достижения финальной маркировки при всех вариантах выполнения. Таким образом, необходимо добавить условие устойчивости бизнес-транзакции, которое вместе с условием (4) будет функционально аналогично условию, задаваемому в [223] для WF-сетей.

Определение 8. Бизнес-транзакцию назовем устойчивой[1], если для любой маркировки не существует последовательность переходов Y такая, что: $\forall M' \in \mathcal{R}(M_0) \nexists Y: M' \overset{Y}{\Rightarrow} M'$. Бизнес-транзакцию назовем *стабильной*, если она удовлетворяет условию (14) сервисной операции и устойчива.

Определение 9. Бизнес-транзакция будет *отказоустойчивой*, когда она стабильная, и/или компенсируемая, и/или восстанавливаемая.

2.5.3. Внутренне-надежная бизнес-транзакция

Пусть $BT = (P, T, A, \Sigma, V, C, G, E, I, H, \Omega)$ – это бизнес-транзакция, тогда BT будем называть *внутренне-надежной бизнес-транзакцией*, если она атомарная, восстанавливаемая, стабильная и отказоустойчивая. Такая бизнес-транзакция способна вместе со своими участниками во время выполнения поддерживать на определенном уровне значения атрибутов, относящихся к ее способности выполняться с должным уровнем качества при установленных условиях за установленный период времени (см. главу 2.1).

Отметим, что надежность бизнес-транзакции, понятие которой дано в этом определении, должно являться труднодостижимым свойством, ввиду следующих причин:

- Операции сервисов могут влиять на среду вне зоны ответственности бизнес-транзакции, например, это касается ручных и полуавтоматических операций, произведенный эффект после которых невозможно отменить. Моделировать такой эффект можно с помощью переходов в позиции, для которых нет выходящих дуг.
- Мощность множества всех композиций операций Q может быть недостаточной для формирования восстанавливаемой или композиционно атомарной

[1] Строго говоря, условие устойчивости может нарушаться, если участники бизнес-транзакции демонстрируют недетерминированное поведение в ответ на одинаковые входные условия. По правилам моделирования сетей Петри такое невозможно, что позволяет аналитически исследовать стабильность бизнес-транзакции, однако на практике, учитывая свойства бизнес-транзакции, этот факт имеет место.

транзакции. Причинами недостаточности Q может выступать небольшое количество операций и неполная их интерфейсная совместимость. Все эти проблемы возникают ввиду финансовых, трудовых и временных ограничений накладываемых на разработку и развитие сервис-ориентированной среды предприятия.

Поэтому скажем, что BT является внутренне-надежной бизнес-транзакцией с ограниченной надежностью (или ограниченно-надежной бизнес-транзакцией), если она устойчивая, частично атомарная и частично восстанавливаемая.

Надежность бизнес-транзакции в определениях, данных нами, не гарантирует, вообще говоря, ее успешное выполнение или откат. Это объясняется тем, что для обеспечения атомарности и восстанавливаемости используются компенсирующие сети, показатели надежности которых могут отличаться от 1. В связи с этим введем понятие порядка надежности.

Пусть $BT = (P, T, A, \Sigma, V, C, G, E, I)$ – это бизнес-транзакция в сервис-ориентированной среде $\mathcal{E}$, тогда BT будем называть внутренне-надежной бизнес-транзакцией k -го порядка, если она устойчивая, атомарная и восстанавливаемая, а используемые в ней восстанавливающие и компенсирующие композиции операций реализуются надежными декомпозирующими сетями вплоть до k-го порядка.

В целях расчетов показателей надежности по результатам моделирования введем несколько определений.

Имитационным прогоном YS бизнес-транзакции BT назовем кортеж, состоящий из последовательности шагов Y, такой, что: $M_0 \overset{YS}{\Rightarrow} M_n$, где n – количество шагов в YS. Набор имитационных прогонов назовем имитационным экспериментом и обозначим за ES. Не исключается возможность параллельного выполнения имитационных прогонов одного эксперимента, что позволяет оценить способность участников бизнес-транзакции выполнять их одновременно.

Назовем имитационный прогон YS с n шагами успешным, если $\exists M_n \in \mathcal{R}(M_0): M_0 \overset{YS}{\Rightarrow} M_n$, где $M_n \in \Omega$; имитационный прогон с отказом, если $\exists M_n \in \mathcal{R}(M_0): M_0 \overset{YS}{\Rightarrow} M_n$, где $M_n = M^{\phi}$; имитационный прогон с аварией, если $\exists M_n \in \mathcal{R}(M_0): M_0 \overset{YS}{\Rightarrow} M_n$, где $M_n = M^{CR}$.

2.6. Расчет показателей надежности бизнес-транзакции

Мы неоднократно отмечали, что на практике не может существовать понятия «абсолютно» надежная бизнес-транзакция. В силу различных ограничений не всегда можно создать компенсирующие или восстанавливающие сети для всех операций, но даже надежность определенного порядка, как мы показали в 2.5, требует разработки восстанавливающих и компенсирующих сетей этого же порядка. Поэтому ранее мы ввели понятие ограниченно-надежной бизнес-транзакции, которая должна быть устойчива и ограниченно-надежна по другим параметрам, и теперь предлагаем подход к точной оценке этих параметров.

Показатели надежности и их граничные значения должны быть определены до начала проектирования бизнес-транзакции. Соблюдение бизнес-транзакцией в течение своей работы границ этих показателей и будет являться отражением ее надежности. В связи с этим формулировка задачи оценки надежности выглядит так.

Пусть задано множество мощностью n показателей и их граничных значений в виде множества двоек $\mathrm{A} = \{(a^b, a^t)_i\}, i = 1..n$, где a^b, a^t – верхние и нижние границы i-го показателя соответственно. А бизнес-транзакция характеризуется набором показателей $W = (w_1, w_2, \dots, w_n)$. Тогда бизнес-транзакция будет называться надежной в том случае, если

$$\forall i: a_i^b \le w_i \le a_i^t. \quad (18)$$

Таким образом, предопределение номенклатуры показателей надежности и их границ перед началом работ над бизнес-транзакцией позволяет сказать, является ли бизнес-транзакция надежной с точки зрения ожидаемых параметров или нет.

В 2.1 мы предложили набор показателей, которые могут характеризовать внутреннюю надежность бизнес-транзакции, а предложенные нами в 2.4 примитивы на языке окрашенных сетей Петри позволяют не только описывать распределенные бизнес-транзакции, функционирующие в сервис-ориентированной среде, но и рассчитывать эти показатели.

Пусть $BT = (P, T, A, \Sigma, V, C, G, E, I, H, \Omega)$ – это бизнес-транзакция, надежность которой рассчитывается в ходе имитационного эксперимента ES. Имитационный эксперимент является набором имитационных прогонов YS, каждый из которых через последовательность n шагов Y переводит бизнес-транзакцию из маркировки M_0 в маркировку M_n, которая является мертвой маркировкой. Множество всех мертвых маркировок имитационного эксперимента состоит из маркировок, означающих успешных финал или отказ BT, т.е. $\forall M_n \in \{M^{CR}, \Omega\}$. Тогда определим показатели

надежности бизнес-транзакции BT как соотношения мощностей соответствующих множеств (см. табл. 5). Некоторые показатели для аварий, например, средняя наработка на аварию, рассчитываются по аналогии с показателями для отказов.

Таблица 5 – Показатели надежности бизнес-транзакции на языке окрашенных сетей Петри

Название	**Формула расчета**
Вероятность безаварийной работы	$X = \frac{\lvert ES' \rvert}{\lvert ES \rvert}$, где $ES' \subseteq ES, \forall YS \in ES' : \exists M_n \in \mathcal{R}(M_0) : M_0 \overset{YS}{\Rightarrow} M_n$, $M_n \in \Omega$
Вероятность безотказной работы	$X = 1 - \frac{\lvert ES^{\Phi} \rvert}{\lvert ES \rvert}$, где $ES^{\Phi} \subseteq ES$, $\forall YS \in ES^{\Phi} : \exists M_n \in \mathcal{R}(M_0) : M_0 \overset{YS}{\Rightarrow} M_n$, $M_n = M^{\Phi}$
Частота отказов	$X = \frac{\lvert ES^{\Phi} \rvert}{\lvert ES \rvert}$, где $ES^{\Phi} \subseteq ES$, $\forall YS \in ES^{\Phi} : \exists M_n \in \mathcal{R}(M_0) : M_0 \overset{YS}{\Rightarrow} M_n$, $M_n = M^{\Phi}$
Частота аварий	$X = \frac{\lvert ES^{CR} \rvert}{\lvert ES \rvert}$, где $ES^{CR} \subseteq ES$, $\forall YS \in ES^{CR} : \exists M_n \in \mathcal{R}(M_0) : M_0 \overset{YS}{\Rightarrow} M_n, M_n \in \Omega$
Наработка на отказ	$X = \frac{\lvert ES' \rvert}{\lvert ES^{\Phi} \rvert}$, где – $ES^{\Phi} \subseteq ES$, $\forall YS \in ES^{\Phi} : \exists M_n \in \mathcal{R}(M_0) : M_0 \overset{YS}{\Rightarrow} M_n$, $M_n = M^{\Phi}$ – $ES' \subseteq ES$, $\forall YS \in ES' : \exists M_n \in \mathcal{R}(M_0) : M_0 \overset{YS}{\Rightarrow} M_n$, $M_n \in \Omega$
Наработка на аварию	$X = \frac{\lvert ES' \rvert}{\lvert ES^{CR} \rvert}$, где – $ES^{CR} \subseteq ES$, $\forall YS \in ES^{CR} : \exists M_n \in \mathcal{R}(M_0) : M_0 \overset{YS}{\Rightarrow} M_n$, $M_n \in \Omega$ – $ES' \subseteq ES$, $\forall YS \in ES' : \exists M_n \in \mathcal{R}(M_0) : M_0 \overset{YS}{\Rightarrow} M_n$, $M_n \in \Omega$
Интенсивность отказов	$X = \frac{\lvert ES^{\Phi} \rvert}{\lvert ES' \rvert}$, где – $ES^{\Phi} \subseteq ES$, $\forall YS \in ES^{\Phi} : \exists M_n \in \mathcal{R}(M_0) : M_0 \overset{YS}{\Rightarrow} M_n$, $M_n = M^{\Phi}$ – $ES' \subseteq ES$, $\forall YS \in ES' : \exists M_n \in \mathcal{R}(M_0) : M_0 \overset{YS}{\Rightarrow} M_n$, $M_n \in \Omega$
Интенсивность аварий	$X = \frac{\lvert ES^{CR} \rvert}{\lvert ES' \rvert}$, где – $ES^{CR} \subseteq ES$, $\forall YS \in ES^{CR} : \exists M_n \in \mathcal{R}(M_0) : M_0 \overset{YS}{\Rightarrow} M_n$, $M_n \in \Omega$ – $ES' \subseteq ES$, $\forall YS \in ES' : \exists M_n \in \mathcal{R}(M_0) : M_0 \overset{YS}{\Rightarrow} M_n$, $M_n \in \Omega$
Вероятность избежания аварий	$X = 1 - \frac{\lvert ES^{CR} \rvert}{\lvert ES^{\Phi} \rvert}$, где

Название	Формула расчета
	– $ES^{\Phi} \subseteq ES$, $\forall YS \in ES^{\Phi}: \exists M_n \in \mathcal{R}(M_0): M_0 \overset{YS}{\Rightarrow} M_n$, $M_n = M^{\Phi}$ – $ES^{CR} \subseteq ES$, $\forall YS \in ES^{CR}: \exists M_n \in \mathcal{R}(M_0): M_0 \overset{YS}{\Rightarrow} M_n$, $M_n \in \Omega$

Продолжение таблицы 5

Название	Формула расчета
Вероятность устранения отказа	$X = \frac{\lvert ES' \rvert}{\lvert ES^{\Phi} \rvert}$, где – $ES^{\Phi} \subseteq ES$, $\forall YS \in ES^{\Phi}: \exists M_n \in \mathcal{R}(M_0): M_0 \overset{YS}{\Rightarrow} M_n$, $M_n = M^{\Phi}$, – $ES' \subseteq ES: (\exists M_n \in \mathcal{R}(M_0), M_n \in \Omega, \forall YS \in ES' \exists \{YS', YS''\}: YS = \{YS', YS''\}: M_0 \overset{YS'}{\Longrightarrow} M^{\Phi} \overset{YS''}{\Longrightarrow} M_n)$.
Вероятность компенсации	$X = \frac{\lvert ES' \rvert}{\lvert ES^{\Phi} \rvert - \lvert ES'' \rvert}$, где – $ES^{\Phi} \subseteq ES: (\forall YS \in ES^{\Phi}: \exists M_n \in \mathcal{R}(M_0): M_0 \overset{YS}{\Rightarrow} M_n$, $M_n = M^{\Phi})$, – $ES'' \subseteq ES: (\exists M_n \in \mathcal{R}(M_0), M_n \in \Omega, \forall YS \in ES'' \exists \{YS', YS''\}: YS = \{YS', YS''\}: M_0 \overset{YS'}{\Longrightarrow} M^{\Phi} \overset{YS''}{\Longrightarrow} M_n)$, – $ES' \subseteq ES: \forall YS \in ES' \exists\ YS' \subseteq YS$ и $\exists q: M_0 \overset{YS'}{\Longrightarrow} M^{\Phi} \overset{q}{\Rightarrow} M_0$, где q – компенсирующая декомпозирующая сеть
Частный уровень атомарности	$X = \frac{\lvert ES \rvert - \lvert ES'' \rvert}{\lvert ES \rvert}$, где $((\exists M_n \in \mathcal{R}(M_0), M_n \in \Omega, ES'' \subseteq ES, \forall YS \in ES'' \exists \{YS', YS''\}): YS = \{YS', YS''\}): M_0 \overset{YS'}{\Longrightarrow} M^{\Phi} \overset{YS''}{\Longrightarrow} M_n$.
Общий уровень атомарности	$X = \frac{\lvert ES'' \rvert + \lvert ES' \rvert}{\lvert ES \rvert}$, где – $ES'' \subseteq ES, \forall YS \in ES'': YS \neq \emptyset, M_0 \overset{YS}{\Rightarrow} M_0$, – $ES' \subseteq ES$, $\forall YS \in ES': \exists M_n \in \mathcal{R}(M_0): M_0 \overset{YS}{\Rightarrow} M_n$, $M_n \in \Omega$
Разрешимость отказов	Расчет не может быть произведен.
Уровень восстанавливаемости	$X = \frac{\lvert ES' \rvert}{\lvert ES' \rvert + \lvert ES'' \rvert}$, где – $ES'' \subseteq ES: (\forall YS \in ES'', \exists \{YS', YS''\}: YS = \{YS', YS''\}): \exists M_n \in \mathcal{R}(M_0), \mathcal{R}(M_n) \cap \Omega = \emptyset: M_0 \overset{YS'}{\Longrightarrow} M^{\Phi} \overset{YS''}{\Longrightarrow} M_n$, – $((\exists M_n \in \mathcal{R}(M_0), M_n \in \Omega, ES' \subseteq ES, \forall YS \in ES' \exists \{YS', YS''\}): YS = \{YS', YS''\}): M_0 \overset{YS'}{\Longrightarrow} M^{\Phi} \overset{YS''}{\Longrightarrow} M_n$.

В обосновании расчетов показателей важную роль играют предельные теоремы теории вероятностей. Легко увидеть, что количество успешных прогонов (или других событий – отказов, компенсаций, восстановлений) $|ES'|$ (для упрощения будем использовать символ m) в ходе одного имитационного эксперимента имеет биномиальное распределение при известном количестве всех прогонов $|ES|$ (будем использовать n) и вероятности безотказной работы $\frac{m}{n}$ бизнес-транзакции. Т.е. вероятность того, что в ходе имитационного эксперимента (или серии выполнения бизнес-транзакции), состоящего из n запусков, количество успешных выполнений $|ES'|$ окажется ровно m раз равна:

$$P(|ES'| = m) = \frac{n!}{(n-m)!m!} * \left(\frac{m}{n}\right)^{km} \left(1 - \frac{m}{n}\right)^{n-m}. \quad (19)$$

При этом, как уже говорилось, мы исходим из предположения, что успех выполнения одной бизнес-транзакции не зависит от другой. Чтобы рассчитать вероятность количество успехов, которые произойдут не более m раз, необходимо суммировать рассчитанные вероятности от $0..m$. Согласно локальной теореме Муавра-Лапласа биномиальное распределение количества отказов (или других показателей) при больших n стремится к нормальному. Таким образом, показатель, который опирается на значение количества успешных событий в каждом имитационном эксперименте, приблизительно может оцениваться с помощью нормального распределения.

В качестве оценки показателя надежности по серии экспериментов мы предлагаем использовать среднее арифметическое: $\frac{1}{n}\sum_{i=1}^{n} X_i$, где X_i – значение показателя в i -м эксперименте. Согласно теореме Чебышева [224] при увеличении n сходится по вероятности к a – среднему арифметическому из их математических ожиданий.

Таким образом, положим, что:

- Количество определенных событий в одном имитационном эксперименте описывается биномиальным распределением.
- Для оценки показателей надежности бизнес-транзакции в целом применимо среднее арифметическое этих показателей, полученных в ходе серии однотипных имитационных экспериментов.
- Для оценки показателей, полученных в ходе серии однотипных имитационных экспериментов, может использоваться предположение, что они имеют нормальное распределение.

Схематично (см. рис. 2) разработанный аппарат моделирования можно представить, как набор уточняющих и ограничивающих примитивов над более обобщенными методами.

Рисунок 2 – Разработанный аппарат моделирования

2.7. Анализ устойчивости бизнес-транзакции с помощью цепей Маркова

Условие стабильности бизнес-транзакции, сформулированное в 2.4 и 2.5, является, на наш взгляд, фундаментальным отличием бизнес-транзакции, выраженной с помощью окрашенных сетей Петри, от любой другой сети. Условие накладывает на сеть ограничение, что любая деятельность должна неуклонно перемещать моделируемую сеть к некоторой финальной маркировке, что финальная маркировка достижима из любой другой маркировки (за исключением случаев отказов и компенсаций). В терминах бизнес-транзакции это означает, что реализация бизнес-транзакции должна приводить к достижению ее целей, к результатам бизнес-процесса, которые представляют определенную ценность для конечного потребителя. При этом, естественно, не должно возникнуть состояния циклов (условие устойчивости) или состояний, которые не позволяют бизнес-транзакции выполниться, «откатиться» или компенсироваться (условие атомарности). Выполнение условия атомарности легко проверить фактом достижения бизнес-транзакции группы домашних маркировок (финальных или начальных). Однако условие устойчивости таким образом проверить труднее.

Например, рассмотрим ситуацию, когда моделируемая бизнес-транзакция с безотказными операциями за n шагов не достигла ни финальной, ни начальной маркировки, хотя известно (например, исходя из анализа построенного графа), что суще-

ствует последовательность шагов, которая приводит бизнес-транзакцию к финальной маркировке. К этой ситуации могли привести 2 причины: последовательность шагов для достижения финальной маркировки меньше n, либо нарушено условие устойчивости и существует последовательность шагов, приводящих к одной и той же маркировке (бесконечный цикл). Другая ситуация предполагает, что финальная маркировка не мертва, то есть после ее достижения бизнес-транзакция продолжает свою активность, что опять является следствием нарушения условия устойчивости, при выполненном условии атомарности.

Исследование моделируемой бизнес-транзакции с помощью цепей Маркова позволяет выявить те операции, маркировки и ситуации, на которые стоит обратить внимания из-за их потенциально опасного, ненадежного поведения [225].

Для определения устойчивости бизнес-транзакции и нахождения потенциально-опасных с точки зрения устойчивости операций и явлений применим цепи Маркова. В первую очередь, необходимо определить систему и состояния, в которых она будет находиться. На наш взгляд, существует 2 подхода ввиду определения бизнес-транзакции, данного в 2.5:

1. Каждая операция бизнес-транзакции может быть отдельным состоянием системы. Цепь Маркова моделирует переходы между этими операциями. Реализация этого подхода возможна только для простой сети Петри, которая следует свойствам безопасности и сохраняемости, и невозможна для более сложных сетей, которые моделируют, например, параллельные операции. Для таких сетей необходимо расширение цепей Маркова, где предполагается, что матрица переходов вероятностей не будет стохастической, т.е. сумма элементов строки будет превышать 1. Однако мы не исключаем возможности частичного использования этого подхода на отдельных группах операции бизнес-транзакции.
2. Каждая маркировка бизнес-транзакции представлена отдельным состоянием системы. Маркировка описывает распределение данных по всем операциям, поэтому является уникальным состоянием системы. Ввиду того, что поведение окрашенной сети Петри строго детерминировано, то можно задать вероятности достижения тех или иных маркировок. Сложность этого метода в том, что с ростом числа операций и «цветов» в окрашенной сети количество всех возможных маркировок нелинейно возрастает, что затрудняет построение матрицы переходных вероятностей.

Несмотря на различие этих подходов, мы покажем, что анализ бизнес-транзакции может потребовать их совместного использования.

Пусть бизнес-транзакция BT может находиться в любой из множества маркировок, достижимых из начальной $\{M_0, \mathcal{R}(M_0)\} = A$. Это множество конечно. Переход между маркировками осуществляется через дискретные промежутки времени. Тогда положим, что последовательность маркировок $M_0, M_1, \dots, M_k$ есть конечная, дискретная, однородная цепь Маркова.

Вероятности перехода между маркировками задается с помощью матрицы $P = \{p_{ij}\}, i, j \leq k$, где строки соответствуют текущей маркировке бизнес-транзакции, столбцы – маркировки на следующем шаге, а на пересечении строк и столбцов вероятность соответствующего перехода.

Ввиду определения бизнес-транзакции соответствующая цепь Маркова ее маркировок является поглощающей. Переформулируем условие устойчивости бизнес-транзакции в терминах этой цепи Маркова.

Бизнес-транзакция BT, смена маркировок которой описывается конечной цепью Маркова, является устойчивой, если множество поглощающих маркировок включает только финальные маркировки и не существует возвратных состояний, отличных от финальных маркировок. Пусть S – множество поглощающих состояний, тогда: $S \subset A$, $S = \Omega$, где A – пространство состояний системы.

Условия атомарности таким способом проверить нельзя, так как это потребует, чтобы начальная маркировка входила во множество поглощающих, но только для любых шагов $n > 1$, что нарушает принцип однородности рассматриваемых цепей Маркова.

Дополнительно проверить отсутствие бесконечных циклов позволяет следующее условие:

Пусть для бизнес-транзакции BT определена фундаментальная матрица $N = (I - Q)^{-1} = \{\mu_{ij}\}$, тогда $\forall i, j: \mu_{ij} \leq 1$.

То есть не должно быть маркировок, которые повторяются более одного раза. Условие имеет ограниченное применение только для сети BT, где не определены сторожевые функции. Следует понимать, что значения фундаментальной матрицы имеют вероятностный характер в виде математического ожидания и, вообще говоря, не гарантируют того, что в очередной, конкретный случай количество переходов в состояние не превысит 1, даже если значение элемента меньше 1. Однако условие позволяет определить потенциально опасные маркировки.

Положим, что за время имитационного эксперимента поглощающее состояние бизнес-транзакцией достигнуто не было, и необходимо определить причины этого.

Так как каждая маркировка – это уникальное распределение данных по позициям бизнес-транзакции, то любое изменение данных (например, приращение или увеличение) в любой позиции приводит к новой маркировке. Постоянная активность какой-либо операции, которая в силу ошибок моделирования или особенностей бизнес-транзакции безостановочно изменяет данные в какой-либо позиции, приведет к неограниченному росту числа маркировок без достижения финальной. С другой стороны, возможно, что в процессе работы БТ необходимая финальная маркировка была достигнута, но она не была мертва и сеть продолжила переход в другие маркировки. Приведем алгоритм, позволяющий выявлять некоторые потенциально-опасные ситуации подобных бесконечных циклов.

Пусть среди пространства состояний A бизнес-транзакции BT, смена маркировок которой описывается конечной цепью Маркова и которая имеет V операций, можно выделить такое подмножество $Z \subset A$, что, начиная с некоторой маркировки n и до k, имеют место следующие положения для $j = i + 1, i = n \dots k$:

- $p_{ij} = 1, \mu_{ij} = 1, D_i[\mu_{ij}] = 0$;
- для всех маркировок $M_n \dots M_k$ существует такое замкнутое множество переходов $T = \{t_v\}: M_i \xrightarrow{t_v} M_j, v = 1..V$;
- если построить цепь Маркова (2-й вариант) для последовательности запусков переходов t_v, а τ_{vl} будет означать переходную вероятность между переходами, и $\sum_{l=1}^{v} \tau_{vl} = 1$. В этом случае, так как все переходы сообщаются только друг с другом, то они составляют один класс эквивалентности и эргодическую цепь Маркова.

Тогда подмножество Z может указывать на маркировки, которые потенциально могут формироваться в условиях бесконечного цикла запусков одних и тех же переходов.

Пусть для множества переходов T, которые участвуют в смене маркировок из множества Z, построена цепь Маркова, где τ_{ij} означает переходную вероятность между переходами сети Петри. Тогда:

1. Если $|T| > 1$, то выделить период класса эквивалентности, в который входят переходы T. Если класс – циклический, то его период может являться периодом оборота бесконечного цикла, и множество переходов T задействованы в этом цикле. Соответствующие алгоритмы и теоремы показаны в [226].

2. Если $|T| = 1$, то новые маркировки генерируются одним переходом, который может являться источником бесконечного цикла.
3. В иных случаях ситуация не может быть однозначно определена. То есть регулярный класс эквивалентности, например, может означать выполнение сложной бизнес-логики или, с другой стороны, сложный бесконечную итерационную процедуру.

Таким образом, отсутствие или наличие потенциально-опасных явлений при выполнении бизнес-транзакции, которые могут повлиять на ее устойчивость, можно показать с помощью следующего алгоритма:

1. Определить все поглощающие маркировки бизнес-транзакции: финальные позиции и позиции отказов (если есть).
2. Провести имитационный эксперимент с моделью бизнес-транзакции для установления вероятностей перехода между состояниями.
3. Построить цепь Маркова для всех маркировок бизнес-транзакции.
4. Построить фундаментальную матрицу и сопутствующие матрицы для проверки условий устойчивости.
5. Если полученная цепь удовлетворяет условиям устойчивости, то бизнес-транзакция может быть устойчивой.
6. Иначе необходимо найти такое подмножество маркировок, которое удовлетворяет условию возможного бесконечного цикла.
   6.1. Если такое множество существует, то необходимо построить цепь Маркова для переходов сети Петри этого множества. Если множество этих переходов составляют один класс эквивалентности с известным периодом, то период может быть периодом оборота бесконечного цикла. Если множество переходов состоит из одного перехода, то он может являться исполнителем бесконечного цикла.
   6.2. В остальных случаях нельзя формально определить источник проблемы устойчивости. На практике это означает чаще всего, что бизнес-транзакция во время имитационного эксперимента просто не успела выполниться до конца.

Кроме прочего, можно воспользоваться полученными по результатам работы алгоритма матрицами для вычисления дополнительных свойств системы. Например, вероятности достижения финальной маркировки из любой другой.

Заметим, что в теории окрашенных сетей Петри существует похожие методы [82]. Графы достижимости схожи с графами цепей Маркова, построенных по маркировкам, а графы строго-связанных компонентов имеют сходство с циклическими

классами и подклассами. Однако принципом построения этих графов является строго-детерминированное поведение моделируемых систем, что не подходит для целей анализа надежности бизнес-транзакции с внедренными стохастическими событиями отказов.

2.8. Методика исследования надежности бизнес-транзакции

В 1.3.4 нами были перечислены основные недостатки аналитических методов расчета надежности, основанных на неформализуемом ручном труде (если говорить о широком поле применения, а не решения отдельной узкой задачи), сложность использования которых возрастает, а точность снижается с ростом сложности систем. Также мы показали, что метод имитационного моделирования, входящий в группу экспериментального подходов к исследованию надежности, будучи универсальным инструментом, может использоваться для исследования надежности безотносительно к структурной сложности системы.

Ввиду простоты представления теории множеств и комплектов с помощью современных средств программирования подготовка аналитических данных и расчет формул из таблицы 6 может осуществляться средствами вычислительной системы. Показатели стандарта ISO 9126 тоже предполагают автоматизированный расчет, но только на базе экспериментов с уже разработанной системой или на базе ее существующего исходного кода. Показатели из таблицы 6 предполагают расчет надежности на основе схемы бизнес-транзакции в виде окрашенной сети Петри, что позволяет предсказать надежность системы еще на стадии ее концептуального проектирования.

Разработанный и описываемый далее (в 2.9) программный комплекс позволяет во время имитационного эксперимента подключать вызываемые из сети сторонние программы, что практически не ограничивает сложность логики выполняемой системы, например, подключать к выполнению эксперимента операторов для выполнения ручных операций или тестировать бизнес-транзакцию только с частью разработанных компонентов.

Как уже отмечалось, в теории простых и окрашенных сетей Петри для анализа свойств построенной модели широко используются такие методы, как дерево достижимости [227], граф достижимости или граф строго-связанных компонентов [82]. Эти методы позволяют исследовать безопасность, ограниченность, сохраняемость и достижимость сети. К сожалению, использование этих методов основано на детерминированном поведении моделируемой системы, где единственный фактор случай-

ности – это равновероятная возможность запуска одного из ряда активных в данный момент времени переходов. Внедренная нами возможность отказа активного перехода во время выполнения сети приводит к нестабильным результатам работы существующих алгоритмов построения и средств анализа этих графов. К примеру, построение графа достижимости для описанной в 3.1 системы штатными средствами комплекса CPN Tools приводит к появлению бесконечных циклов в программе. Поэтому для исследования бизнес-транзакции нами предполагается использовать метод статистического моделирования, где показатели надежности системы оцениваются путем проведения множества имитационных экспериментов с моделью, случайные отказы элементов которой имеют ту же вероятность, что и отказы реальных объектов, которые отображаются этими элементами.

Так как используемый нами аппарат окрашенных сетей Петри не моделирует время или длительность отдельных событий, то практическое использование методов, где фактор времени играет важную роль при оценке работы модели, затруднено. Например, это связано с рядом законов распределения случайных величин, описывающих количество отказов элемента за отдельный промежуток времени, или затрудняет внедрение в модель бизнес-транзакции генератора потока событий (отказов, заявок и пр.), зависящих от времени. Учет фактора времени открывает широкие перспективы использования теории массового обслуживания и тех областей теории надежности, где отказы зависят от времени непрерывной работы элементов. Реализация подобных механизмов возможна, например, путем добавления к модели бизнес-транзакции элементов «Временных» сетей Петри (англ. Timed nets), что является перспективным направлением развития предложенного подхода к моделированию бизнес-транзакции.

Предлагаемая методика оценки надежности бизнес-транзакции основана на следующих положениях:

- Бизнес-транзакция решает задачи интеграции процессно-ориентированных информационных систем и может осуществляться в зонах ответственности разных исполнителей (см. главу 1.1.6).
- Бизнес-транзакция обладает свойствами бизнес-процесса, длительной транзакции и сервисной композиции. Поэтому оценка ее надежности должна производиться с учетом этих трех точек зрения (см. главу 2.1).
- Бизнес-транзакция имеет явно определенную цель, поэтому ее выполнение должно стремиться к достижению цели (см. главу 1.3.3).

- Бизнес-транзакция отображает операции и объекты реального мира, поэтому количество используемых ею ресурсов должно быть ограниченно.
- Понятие внутренней надежности бизнес-транзакции позволяет отделять надежность собственно бизнес-транзакции от надежности бизнес-транзакции вместе с ее участниками и от влияния дальнейших последствий их поведения.
- Внутренняя надежность бизнес-транзакции качественно определяется по ее структуре и отдельным динамическим характеристикам помощью оценки атомарности, устойчивости и восстанавливаемости (см. главу 2.5).
- При наличии возможности тестирования или моделирования количественный уровень внутренней надежности можно определить, рассчитав показатели атрибутов надежности (см. таблицу 5).

Опишем методику с использованием алгоритма оценки надежности бизнес-транзакции, составленным с учетом ранее полученных выводов. Методика частично автоматизируется с помощью разработанного программного комплекса и включает следующие операции:

1. Идентификация и отделение бизнес-транзакции как системы от предметной области. Формирование целей существования системы, определение ее границ и хода функционирования.
2. Анализ структуры и состава бизнес-транзакции. Определение участников и процессов, которые они выполняют. Определение связей между процессами. Определение зон ответственности хореографии и оркестровки.
3. Формулировка и анализ проблемы оценки надежности исследуемой системы. Определение взаимосвязи надежности с другими характеристиками. Оценка разрешимости проблемы.
4. Оценка возможностей повышения надежности по процессам и в целом для бизнес-транзакции путем введения восстанавливающих и компенсирующих связей: выявление технологии интеграции участников и определение характера связанности.
5. Определение целевых качественных и количественных показателей надежности. Определение нормативных значений этих показателей, если требуется. Определение множества исследуемых операций и характеристик их надежности.
6. Первичное конструирование модели бизнес-транзакции с помощью окрашенных сетей Петри и описание бизнес-транзакции теоретико-множественной формулой: выделение наборов цветов, описание перемен-

ных и функций; определение множества участников; разработка схемы транзакции (определение множества позиций и переходов) с учетом шаблонов оркестровки и хореографии; определение инициализирующих (начальных) данных для позиций бизнес-транзакции; определение начальной маркировки, множества финальных и промежуточных маркировок.

7. Если сложность бизнес-транзакции невысока, то возможно аналитическое исследование модели: изучение ее атомарности, восстанавливаемости, компенсируемости, стабильности. Определение внутренней надежности по схеме бизнес-транзакции. Если данных достаточно, то СТОП.
8. Инициация нового эксперимента: загрузка файла со схемой в среду имитатора, определение названия эксперимента, количества внутренних запусков, приблизительного количества срабатываний переходов до завершения бизнес-транзакции, типа анализа надежности (анализ надежности в целом, анализ чувствительности к операциям).
9. Описание сетей, найденных программным комплексом на загруженной схеме, и позиции успешного завершения бизнес-транзакции. Выбор исследуемых процессов, задание характеристик их надежности и методов повышения надежности через восстанавливающие или компенсирующие сети.
10. Проведение имитации работы бизнес-транзакции.
11. Анализ результатов имитационного моделирования: показателей надежности по заданным параметрам системы, чувствительности надежности системы к надежности исследуемых операций, анализ устойчивости с помощью цепей Маркова.

Если цели оценки надежности за одну итерацию не достигаются, то в схему транзакции вносятся изменения и этапы 8-11 повторяются. Результаты апробации этой методики для анализа надежности были показаны, например, в [228]. Этапы 6, 8-11 автоматизируются разработанным программным комплексом.

Предполагается, что проверка модели на языке окрашенных сетей Петри на адекватность, непротиворечивость и работоспособность осуществляется самим исследователем в процессе разработки схемы. Так как очень часто процессы схематично отображаются в виде последовательностей преобразования и передачи ресурсов (материальных или информационных), то аппарат сетей Петри позволяет напрямую «отображать» такую схему. Трудности у исследователя возможны только при реали-

зации сложной логики преобразования ресурсов на схеме, но для этого существует возможность подключения внешних программных средств. Кроме того, схема бизнес-транзакции может являться тем проектом, по которому она может быть построена в будущем, и вопрос проверки адекватности снимается. Например, как отмечается в [229] оценка адекватности моделей проектируемых объектов или математических моделей таких объектов [105] весьма проблематичная задача, выполняемая экспертным путем.

2.9. Описание программного комплекса

Разработанный программный комплекс (далее программный комплекс или ПКБТ, на англ. языке – «CPN Reliability Simulator») предназначен для оценки надежности проектируемой бизнес-транзакции, функционирующей в сервис-ориентированной среде. ПКБТ позволяет автоматизировать решение всех 4 типов задач оценки надежности бизнес-транзакций, выделенных нами в 1.3.6.

ПКБТ решает задачу оценки общей надежности проектируемой бизнес-транзакции любой сложности при условии, что информация о надежности / работоспособности ее участников известна. Как мы уже говорили ранее, при наличии подобной информации о надежности системы можно рассчитать аналитически, однако по мере усложнения системы сложность и точность применения любых математических методов существенно снижается. ПКБТ решает проблему методом имитационного моделирования.

Методической основой ПКБТ является алгоритм оценки надежности бизнес-транзакции, описанный в 2.9.

Программный комплекс состоит из нескольких взаимосвязанных частей (см. рис. 3) [230]:

Реализованы автором:

- Интерфейс пользователя реализован на языке HTML, который предоставляет возможность логического форматирования элементов интерфейса, языка CSS для стилевого оформления и ряд программ на языке JavaScript для интерактивного взаимодействия с пользователем и реализации асинхронного взаимодействия с сервером приложений. Таким образом, интерфейсная часть должна работать в среде, которая интерпретирует эти языки, например, в веб-браузере.
- Сервер приложений написан на языке Perl. Он обрабатывает запросы пользователя, анализирует данные от симулятора и проводит их статистическую об-

работку, осуществляет работу с сервером баз данных. Сервер приложений способен в одном экземпляре одновременно обрабатывать запросы нескольких клиентов-пользователей, однако сервер баз данных должен быть одним.

- Вычислительные библиотеки необходимы для анализа результатов имитационного моделирования. Они содержат функции для расчета статистических показателей, проведения дисперсионного анализа, проверки разнообразных гипотез, генерирования критических значений для разных законов распределения. Важную часть составляют функции для использования цепей Маркова для построения матриц переходных вероятностей, приведений их к каноническому виду и нахождения фундаментальной матрицы.
- Сервер баз данных представлен свободной реляционной СУБД PostgreSQL. Сервер проводит хранение и обработку информации о пользователях ПКБТ, их прошлых и текущих экспериментах и загруженных в ПКБТ схем сетей.

Рисунок 3 – Состав программного комплекса

- Контроль симулятора – это структура, состоящая из двух отдельно запускаемых программ. Первая программа написана на языке Perl, вызывает библиотеки и функции сервера приложений и используется для запуска второй части и контроля работы симулятора в целом. Вторая программа написана на языке

Java, предназначена для инициализации и запуска симулятора, а для этого использует библиотеку ACCESS/CPN.

Реализованы работниками Технического университета Эйндховена:

- ACCESS/CPN – это библиотека в составе CPN Tools, написанная на языке Java разработчиком симулятора для управления им из сторонних программных систем. По сложности и объему программного кода, а также трудностям работы с ней с точки зрения доработки она сравнима с остальными частями пакета CPN Tools [231].
- Среда отрисовки схем реализована в пакете CPN Tools. Этот пакет распространяется под свободной лицензией GNU GPL версии 2. Среда отрисовки является интерфейсом пользователя для CPN Tools и написана на объектно-ориентированном языке BETA. Среда позволяет создавать схемы окрашенных сетей Петри, поддерживает временные и иерархические сети и позволяет проводить имитацию работы сети через управление симулятором CPN Tools. Подробное описание пакета приводится в [82] и на официальном сайте [232].
- Симулятор входит в состав пакет CPN Tools и реализован на языке Standard ML. В симулятор встроен компилятор языка, что позволяет описывать на схеме сети подпрограммы на этом языке и выполнять их в процессе имитации. Что позволяет реализовать сложную логику поведения участников БТ, например, осуществлять вызов сторонних программ путем установления сетевых соединений.

Необходимость вынесения функций контроля над работой симулятора вне сервера приложений и разделения этих функций на две части продиктовано практической целесообразностью. Во-первых, множественные эксперименты с симулятором и анализ его исходного кода показали, что это крайне нестабильное приложение, требующее монопольного запуска в операционной системе. Причем в зависимости от множества явных (например, ошибки в схеме процесса) и неявных факторов (например, уровень загрузки операционной системы, поведение пользователя и пр.) симулятор может прекратить свою работу или войти в тупиковую ситуацию, что не позволяет запускать его напрямую из сервера приложений. Причем процесс, запускающий симулятор, ввиду блокировки потоков ввода/вывода также блокируется на неопределенное время. Поэтому для запуска и контроля над работой процесса симулятора необходим сторонний процесс, который можно «уничтожить» без потери работоспособности сервера приложений. Этот процесс и составляет первую часть контроля симулятора. Во-вторых, несмотря на то, что симулятор распространяется под

лицензией GNU GPL версии 2, он фактически не имеет документации для программиста. Исходный код симулятора крайне скудно комментирован и содержит множество неявных и дублирующихся фрагментов, поэтому автор программы рекомендует использовать библиотеку ACCESS/CPN, которая написана на языке Java. Эту библиотеку подключает и использует вторая часть контроля симулятора, написанная на языке Java. Задача этой части – запуск симулятора с определенными параметрами работы. Между частями контроля симулятора взаимодействие осуществляется посредством прямого запуска процесса первой частью второй. В целом эти проблемы эксплуатации, сопровождения и доработки исходного кода едины для всех частей пакета CPN Tools (среды отрисовки схем, библиотеки ACCESS/CPN и симулятора), и это чрезвычайно затрудняет процесс их использования.

Как можно отметить, все части ПКБТ реализованы с помощью свободного программного обеспечения [233], что позволяет использовать его в любой операционной среде. В целом ПКБТ является распределенной программной системой, которая способна функционировать на разных операционных системах и разном вычислительном оборудовании. С этой целью ПКБТ разделен на 4 самостоятельных части: среда отрисовки, интерфейс, сервер баз данных, сервер приложений с симулятором и контролем симулятора, а также внешними программами. Каждая часть может взаимодействовать с другими посредством сетевых коммуникаций, например, через сеть интернет. В целях масштабирования существует техническая возможность расширить возможности контроля симулятора до параллельных операций на множестве симуляторов, запущенных на множестве вычислительных систем.

Интерфейс вместе с сервером приложений и сервером базы данных представляет трехзвенное клиент-серверное приложение [15], которое может функционировать и без доступа к CPN Tools, например, с целями анализа ранее проведенных имитаций.

Кроме пакета CPN Tools для своей работы ПКБТ использует также фреймворки jQuery (открытая лицензия MIT) и Highcharts (бесплатная лицензия Creative Commons Attribution-NonCommercial 3.0 License).

Все вышеперечисленное позволяет реализовать ПКБТ в виде веб-сайта, где со стороны клиентов сайта осуществляются операции по отрисовке схем и управления процессом имитации через интерфейс сайта, а со стороны серверной части выполняется работа с данными, управление симулятором и взаимодействие со сторонними программами. Веб-сайт может быть размещен как в сетевой среде предприятия, учебного заведения, так и на любом сервере сети интернет. Способность сервера приложений и базы данных обрабатывать множество клиентов одновременно позво-

ляют использовать ПКБТ в учебном процессе или для обслуживания нескольких проектировщиков бизнес-транзакций сразу.

Основным преимуществом ПКБТ перед распространенными средствами имитационного моделирования является:

- Ориентация на предметную область. ПКБТ предназначен для изучения надежности бизнес-процессов любой сложности, безотносительно к их свойствам. Конечно, он позволяет оценивать бизнес-процесс как бизнес-транзакцию и обладает инструментами, оценивающими эффективность использования сервис-ориентированных технологий, таких как взаимозаменяемость и компонуемость сервисов, но способен исследовать и простые бизнес-процессы, помогать исследователю их выявлять «бутылочные горлышки» и оценивать эффективность управления.
- Легкость освоения. Исследователю нет необходимости досконально изучать язык моделирования, дополнительные библиотеки, способы генерации сигнала. Графической редактор на базе окрашенных сетей Петри прост в изучении и легок в употреблении.
- Способность оперировать сложными наборами данных. Большинство средств имитационного моделирования позволяется манипулировать простыми сигналами, имеющими числовые значения, лишь некоторые средства позволяют проводить исследование нескольких простых сигналов одновременно. ПКБТ же используется все преимущества окрашенных сетей Петри, которые позволяют манипулировать сколь угодно сложными структурами данных, проводя с ними различные операции.
- Неограниченная сложность модели. Симулятор CPN Tools, который использует ПКБТ, позволяет определять на схеме модели вызов сторонних программ, написанных на любых языках программирования для реализации сложной логики обработки данных. Таким образом, теоретически сложность модели бизнес-процесса, который оценивает ПКБТ, ничем не ограничена.
- Возможность проводить задействовать в эксперименте ресурсы бизнес-транзакции. Взаимодействие со сторонними программами во время имитационного эксперимента и обмен данными с ними позволяет включать в эксперимент ручные операции или проводить нагрузочное тестирование оборудования, выполняющее операции бизнес-транзакции. Это же свойство позволяет, например, исследовать надежность частично-готовой системы, тестируя уже разработанные компоненты во время имитации.

Сравнение результатов работы ПКБТ и расчетов с помощью популярных методов теории надежности даны в Приложении Б.

Выводы по главе 2

В первой главе было показано, что, ввиду того, что бизнес-транзакция представляет собой сложную систему, функционирующую в распределенной среде, то для анализа ее надежности может быть стандартные методы теории надежности могут быть недостаточно эффективными.

Поэтому было предложено рассматривать бизнес-транзакцию в сервис-ориентированной среде как систему с тройственной природой: как транзакции, композиции сервисов и бизнес-процесс, обладающий целью, результатом, потребляемыми ресурсами, свойствами физической необратимости, стоимости и времени выполнения. Это позволило использоваться методики анализа надежности из трех областей и выработать набор показателей надежности, которые позволяют оценивать бизнес-транзакцию.

Для решения задачи формального описания бизнес-транзакции и расчета показателей надежности был разработан специальный аппарат моделирования бизнес-транзакции, который основан на теории окрашенных сетей Петри. Преимуществом разработанного аппарата является возможность формальной оценки показателей надежности бизнес-транзакции, формулы расчета которых также были предложены.

Мы показали, что одно из важных свойств надежной бизнес-транзакции – ее устойчивость не может быть определена с помощью имитационных испытаний ее модели, поэтому был разработан специальный алгоритм, использующий теорию цепей Маркова.

Эти выводы опираются на следующие результаты, полученные в третьей главе:

1. Разработано определение надежности бизнес-транзакции. Бизнес-транзакция рассмотрена с трех точек зрения надежности: как транзакция, композиция сервисов и бизнес-процессов. Предложены атрибуты надежности. Дано определение внутренне-надежной бизнес-транзакции. Рассмотрены показатели надежности. Показано, что при стабильных условиях работы надежность бизнес-транзакции зависит только от атомарности, отказоустойчивости и восстанавливаемости. Предположено, что восстанавливаемость бизнес-транзакции в СОА можно повысить путем использования имманентного свойства замены сбойной службы другой, аналогичной по функциональности.

2. В качестве базового инструмента моделирования бизнес-транзакции выбраны сети Петри. Дан краткий конспект теории комплектов, окрашенных сетей Петри, окрашенных сети Петри, сопутствующих определений и свойств этой сети. Проведено описание иерархических окрашенных сети Петри, WF-сетей и сервисных сети.
3. Предложено рассмотреть сервис как множество конечных точек. Каждая конечная точка определена совокупностью операций. Предложено отображать каждую конечную точку на отдельный переход окрашенной сети Петри. Показано, почему иерархическое окрашенные сети Петри не подходят для моделирования бизнес-транзакции в сервисной среде.
4. Введено определение сервисной операции в терминах окрашенной сети Петри, предусловия и постусловия операции. Дано определение интерфейсно-совместимости операций. На основе сервисной операции введено понятие сервисной композиции.
5. Введено понятие декомпозирующей сети различного порядка. Показано отличие между атомарной и композитной операцией. Показаны преимущества введенного аппарата моделирования перед существующими методиками на базе сетей Петри. Таким образом, разработан аппарат для моделирования и анализа сервис-ориентированных систем.
6. Введено определение бизнес-транзакции в терминах сервисных операций и окрашенных сетей Петри. Определено условие, при котором бизнес-транзакцию можно считать сервисной композицией. Дано понятие восстанавливаемой операции и восстанавливаемой бизнес-транзакции. В терминах сервисных операций дано понятие атомарной и атомарной по структуре бизнес-транзакции. Введено понятие компенсируемого состояния. Введено понятие устойчивой бизнес-транзакции. Внутренне-надежная бизнес-транзакция определена как атомарная, восстанавливаемая и устойчивая. Введено понятие внутренне-надежной k-го порядка и частично внутренне-надежной бизнес-транзакции. Таким образом, разработан инструментарий для моделирования бизнес-транзакций всех типов.
7. Для уверенной оценки устойчивости бизнес-транзакции предложено использовать аппарат цепей Маркова. Рассмотрено два пути приложения цепей Маркова к анализу устойчивости. Предложен алгоритм оценки устойчивости бизнес-транзакции, предполагающий построение и анализ цепи Маркова для маркировок и переходов бизнес-транзакции.

8. Сформулирована задача оценки надежности бизнес-транзакции. Предложены формулы для расчета показателей надежности в терминах разработанного аппарата моделирования.
9. Описана методика анализа надежности бизнес-транзакции на основании всех результатов, полученных ранее.
10. Описан программный комплекс, его структура, преимущества и недостатки по сравнению с другими системами имитационного моделирования.

На основании формальных примитивов, описывающих элементы сервис-ориентированной архитектуры и бизнес-транзакций, полученных в 2.3, была обоснована возможность моделирования бизнес-транзакций на языке окрашенных сетей Петри. Полученные в 1.3.6 обоснования преимуществ имитационного моделирования при расчете надежности и показатели надежности из 2.2 позволили провести разработку методики и программного комплекса, способного отображать бизнес-транзакцию на базе сетей Петри, проводить ее имитационное моделирование и осуществлять анализ ее надежности.

Поэтому задачами 3-й заключительной главы, являются:

- Описание порядка работы с программным комплексом с учетом предлагаемой методики.
- Описание этапов исследования проектируемой бизнес-транзакции для существующего предприятия с целью повышения ее надежности.

3. Апробация методики и ПО исследования надежности бизнес-транзакции

3.1. Пример использования методики исследования надежности бизнес-транзакции

Рассмотрим процесс применения предлагаемой методики на примере простой бизнес-транзакции проведения оплаты. Предположим, что исследование надежности осуществляется с точки зрения компании (назовем ее компания А), оплату услуг которой клиент может производить с помощью платежного терминала (принадлежащий компании Б). Целью анализа является выявление того, насколько надежна эта процедура.

1. Определение целей и границ существования бизнес-транзакции.

Процесс функционирования бизнес-транзакции может описать так (см. рис. 4): клиент совершает оплату через платежный терминал, терминал уведомляет об это сервер платежной компании Б, тот уведомляет сервер компании А. Сервер компании А уведомляет клиента о приеме платежа с помощью сервера сообщений, принадлежащего компании В.

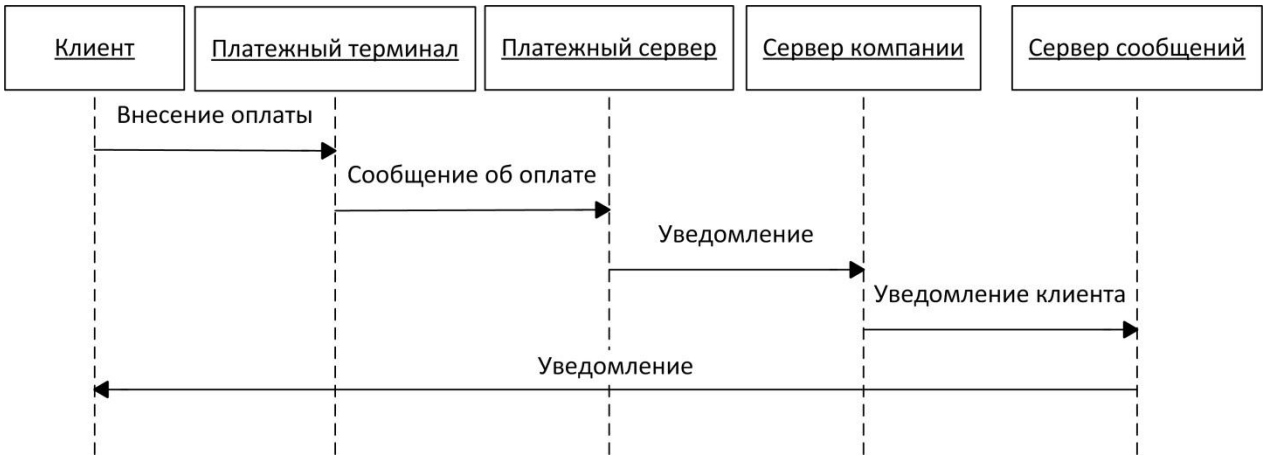


Рисунок 4 – Процесс функционирования бизнес-транзакции

Целью работы бизнес-транзакции является успешное уведомление сервера компании А о проведенном платеже. Какие-либо операции в бизнес-транзакции, которые не приводят к уведомлению сервера, не имеют смысла ни для одного участников БТ.

Отметим, что в качестве примера мы рассматриваем упрощенный, абстрактный пример бизнес-транзакции, лишенный многих необходимых деталей. Для анализа надежности подобной простой системы, если бы она существовала в реальности, возможно целесообразно будет применение стандартных методов теории надежности.

2. Анализ состава и структуры системы.

В бизнес-транзакции принимают участие клиент, платежный терминал, платежный сервер, сервер компании, сервер сообщений. Каждый участник реализует одну или несколько операций (см. табл. 6). Все операции являются вызываемыми по запросу.

Таблица 6 – Операции участников

Участник	Операции
Клиент	Прием информации о поступлении платежа
Платежный терминал	Прием платежа, отправка информации о платеже
Платежный сервер	Прием информации о платеже, отправка уведомления о платеже
Сервер компании	Прием уведомления о платеже, зачисление платежа на счет клиента, отправка сообщения для передачи клиенту
Сервер сообщений	Прием сообщения для клиента, передача сообщения клиенту

В БТ задействовано несколько независимых участников, каждый из которых обладает состоянием, поэтому вся она находится в зоне хореографии (см. рис. 5). А так как бизнес-транзакция рассматривается с точки зрения компании А, то в пределах ее работы функционирует шаблон оркестровки, а оркестратором является ее сервер.

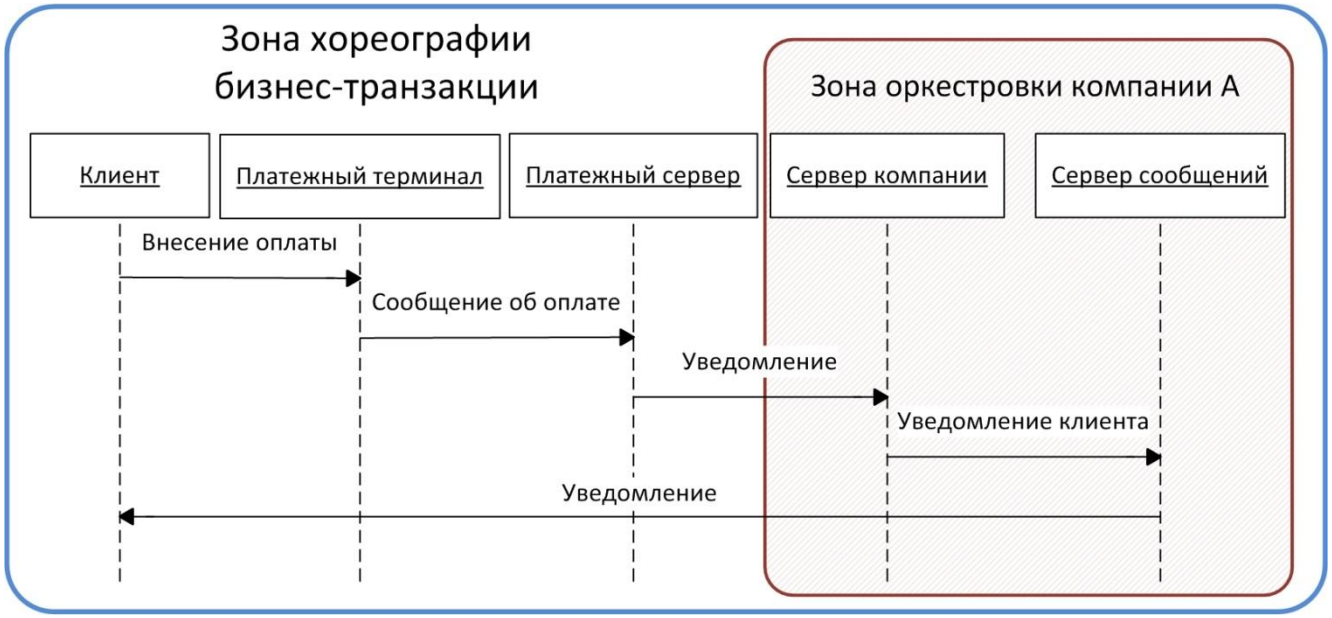


Рисунок 5 – Зоны оркестровки и хореографии

3. Формулировка задачи оценки надежности

В процессе анализа надежности необходимо выявить значения основных показателей надежности, а также определить степень влияния каждой операции и участника на общую надежность системы.

4. Оценка возможностей повышения надежности.

Клиент взаимодействует с платежным терминалом напрямую. Все операции остальных участников построены с использованием веб-сервисов. Таким образом, характер связанности: слабый.

5. В процессе анализа необходимо установить значения показателей: вероятность безотказной и безаварийной работы, наработка на отказ и аварию, вероятность устранения аварии и отказа, вероятность компенсации, разрешимость отказов, общий уровень атомарности бизнес-транзакции.

Предварительное исследование надежности участников системы, а также информация из соглашений об уровне обслуживания (англ. SLA) позволили выявить следующую вероятность отказа операций (см. табл. 7):

Таблица 7 – Надежность отдельных операций

Участник	**Операции**	**Вероятность отказа**
Клиент	Прием информации о поступлении платежа	0,13
Платежный терминал	Прием платежа	0,013
	Отправка информации о платеже	0,0027
Платежный сервер	Прием информации о платеже	нет данных
	Отправка уведомления о платеже	нет данных
Сервер компании	Прием уведомления о платеже	0,008
	Зачисление платежа на счет клиента	0,001
	Отправка сообщения для передачи клиенту	0,001
Сервер сообщений	Прием сообщения для клиента	0,008
	Передача сообщения клиенту	0,045

6. Конструирование модели бизнес-транзакции с помощью окрашенных сетей Петри

Пусть бизнес-транзакция описывается сетью Петри $BT = (P, T, A, \Sigma, V, C, G, E, I, H, \Omega)$, где:

- T – множество операций, $T = \{Get\ pay, Send\ Info, Get\ info, Send\ notif, Get\ notif, Account, Send\ msg, Get\ msg, Send\ sms, Get\ sms\}$, что означает соответственно прием платежа, отправка информации о платеже, прием информации о платеже, отправка уведомления о платеже, прием уведомления о платеже, зачисление платежа на счет клиента, отправка сообщения для передачи клиенту, прием сообщения для клиента, передача сообщения клиенту, прием сообщения о поступлении платежа. $|T| = 10$.
- P – множество позиций бизнес-транзакции, среди которых выделяются промежуточные P_m и стабильные позиции P_f такие, что $P = P_m \cup P_f$; Стабильные позиции можно разделить на входные P_{in} и выходные P_{out}, где $\bullet P_{in} \cap P_{out} \bullet$

$= \emptyset$. Входная позиция представлена информацией от клиента для терминала: $P_{in} = \{client\}$, выходная информация формирует данные о счете клиента и уведомление для него: $P_{out} = \{account\ info, sms\}$. Промежуточные позиции связывают переходы между собой. $|P| = 12$.

- A – множество направленных дуг, связывающих пару «переход и позиция». $|A| = 21$.
- Σ – множество непустых типов данных, используемых в транзакции. $\Sigma = \{INT \times INT, INT \times INT \times STRING\}$. То есть для моделирования бизнес-транзакции будут использоваться следующие типы данных: целочисленные, строковые и производные от них – структуры из этих типов данных. Для удобства тип $INT \times INT$ назван *cl_data* (данные о клиенте), а $INT \times INT \times STRING$ – *cl_msg* (сообщение).
- V – множество типизированных переменных, $V = \{cid: INT, info: STRING, data: cl\_data, msg: cl\_msg\}$.
- $C(p) = \begin{cases} INT \times INT, \text{если } p \in \{Get\ pay, Send\ Info, Get\ info, \\ Send\ notif, Get\ notif, Account\} \\ INT \times INT \times STRING, \text{если } p \in \{Send\ msg, Get\ msg, \\ Send\ sms, Get\ sms\}. \end{cases}$
- $G(t) = true, \text{если } t \in T.$
- $E(a) = \begin{cases} 1`(data), \text{если передают данные о платеже} \\ 1`(msg), \text{если передает уведомление о платеже} \\ 1`(data, info), \text{если создается сообщение клиенту}. \end{cases}$
- $I(p) = M_0(p) = \begin{cases} 1`(\text{номер клиента, сумма}), \text{если } p = client \\ \emptyset, \text{ если } p \neq client. \end{cases}$
- H={клиент, платежный терминал, платежный сервер, сервер компании, сервер сообщений}.
- $\Omega(p) = \begin{cases} 1`(\text{номер клиента, сумма}), \text{если } p = account\ info \\ \text{любое, если } p \neq account\_info. \end{cases}$, то есть финальной маркировкой бизнес-транзакции является достижение данных позиции *account_info.*

Отобразим эту бизнес-транзакцию в виде графа (см. рис. 6). Как можно заметить, все аналитически описанные элементы присутствуют на графе.

7. Аналитическое исследование модели.

Отметим, что для данной бизнес-транзакции не определены восстанавливающие или компенсирующие сети. Что позволяет утверждать об отсутствии у нее свойств атомарности, восстанавливаемости или компенсируемости.

8. Инициация имитационного эксперимента.

Инициация имитационного эксперимента состоит в загрузке информации о сети Петри в имитационный комплекс ПКБТ. Подробно начала эксперимента описан в Приложении А.

9. Подготовка имитационного эксперимента.

В процессе подготовки эксперимента данные из таблицы 7 переносятся в ПКБТ, указывается финальная маркировка и определяется назначение найденных сетей в модели БТ.

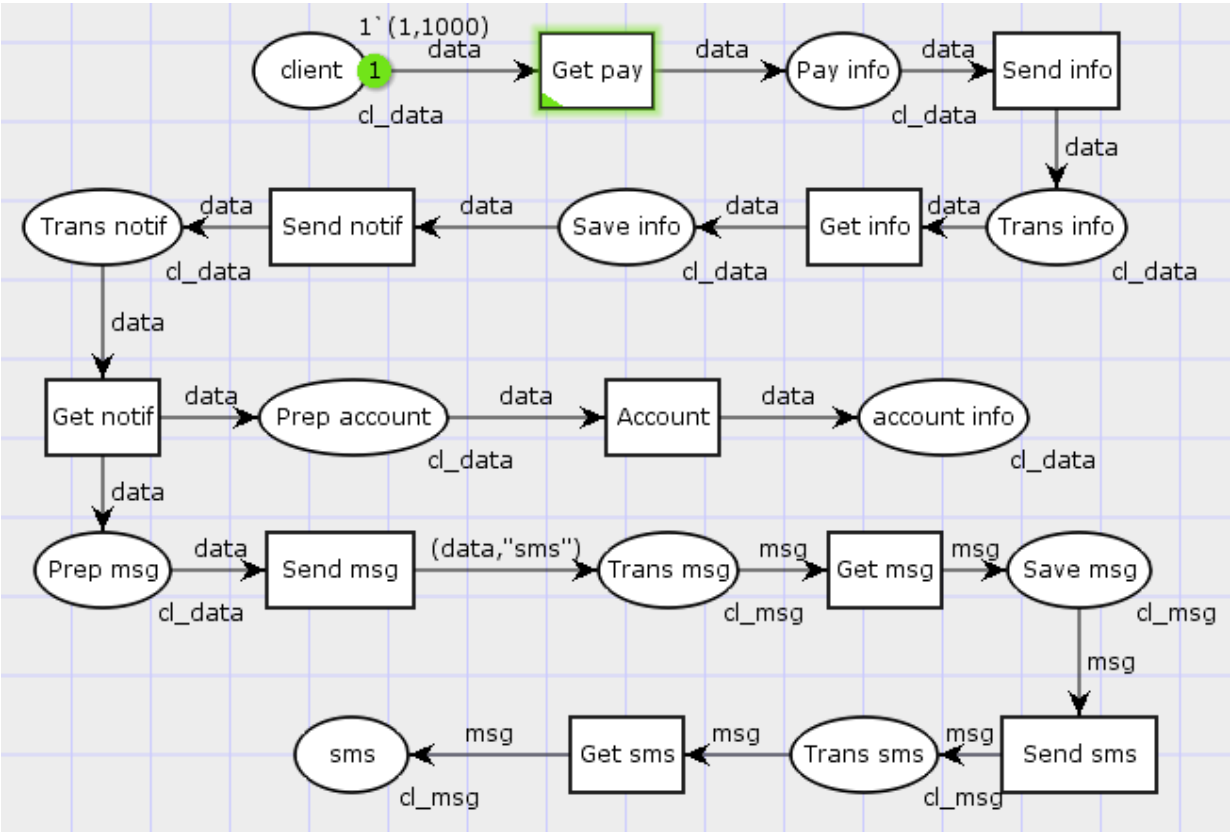


Рисунок 6 – Граф бизнес-транзакции

10. Проведение имитации работы бизнес-транзакции.

Процесс полного анализа надежности потребовал порядка 720 тысяч имитационных прогонов бизнес-транзакции. В процессе анализа исследовалась общая надежность БТ и влияние на надежность отдельных операций.

Аналитически опишем фрагмент работы ПКБТ при выполнении анализа.

Сначала бизнес-транзакция инициализируется в начальной маркировке с любым случайным значением переменной *data*, например, 1 и 10000 (идентификатор клиента и сумма платежа соответственно):

$$M_0(p) = \begin{cases} 1`(1,10000), & \text{если } p = client \\ \emptyset, & \text{в другом случае} \end{cases}$$

Далее маркировка M_1 достигается путем выполнения перехода *Get pay*:

$M_1(client) = (M_0(client) - -E(client, Get\ pay)) + +E(Get\ pay, Pay\ info)$

В случае наличия связей перехода с несколькими позициями сменилась бы маркировка каждого. Естественно, в результате перехода данные из позиции *client* будут перемещены в *Pay info*: $M_1(client) = \emptyset$, $M_1(Pay\ info) = 1`(1,10000)$, т.е. $M_1(p) = \emptyset \forall p \neq Pay\ info$.

Подобный переход из одной кодировки в другую мы называем шагом Y и обозначаем его как $M_0 \xrightarrow{Y} M_1$ (см. главу 2.3).

Через n шагов бизнес-транзакция в случае успешного выполнения достигнет маркировки M_n: $M_0 \xrightarrow{Y_1} M_1 \xrightarrow{Y_2} M_2 \dots M_{n-1} \xrightarrow{Y_n} M_n$. Пусть при этом $M_n(account\ info) = 1`(1,10000)$. Заметим, что переход *Get notif* задает параллельную ветвь выполнения бизнес-транзакции, и какая-то из дальнейших позиций этой ветви будет иметь непустую маркировку M_n. Достижение пакета данных позиции *account info* считается успешным завершением бизнес-транзакции, так как:

$$M_n(p) = \begin{cases} 1`(1,10000), & \text{если } p = account\ info \\ \emptyset, \text{если } p \in \{client, Pay\ info, Trans\ info, Save\ info, Trans\ notif\} \\ \text{любое, если } p \in \{Prep\ msg, Trans\ msg, Save\ msg, Trans\ sms, sms\} \end{cases} \Rightarrow$$

$\Rightarrow M_n(p) \in \Omega$.

Набор шагов $YS = \{Y_1, Y_2, \dots, Y_n\}$ – имитационный прогон. Если $M_n(p) \in \Omega$, то имитационный прогон считается успешным. Если же в процессе выполнения бизнес-транзакции произошел отказ, бизнес-транзакция остановилась или перешла в неверное состояние, из которого не смогла выйти, то имитационный прогон завершился аварией. Множество маркировок аварий названо нами M^{CR} в 2.5.3. Ситуация, в которой нормальный ход выполнения бизнес-транзакции был нарушен, однако после этого бизнес-транзакция смогла успешно завершиться, является отказом. Множество маркировок отказа определяется как M^{ϕ}, при этом имитационный прогон $YS = \{Y_1, Y_2, \dots Y_m, \dots, Y_n\}$: $M_0 \xrightarrow{Y_1} M_1 \xrightarrow{Y_2} M_2 \dots M_{m-1} \xrightarrow{Y_m} M_m \dots M_{n-1} \xrightarrow{Y_n} M_n, M_n \in \Omega, M_m \in M^{\phi}$ назовем имитационным прогоном с отказом.

Имитационный эксперимент ES состоит из множества прогонов. Предположим, что в процессе имитационного эксперимента, состоящего из 50 прогонов, 20 прогонов завершилось с отказом, из которых 10 – с аварией. Тогда, например, некоторые показатели рассчитываются так:

Вероятность безаварийной работы $X = \frac{|ES'|}{|ES|}$, где $ES' \subseteq ES, \forall YS \in ES' : \exists M_n \in \mathcal{R}(M_0) : M_0 \overset{YS}{\Rightarrow} M_n$, $M_n \in \Omega$, т.е. показатель определяется отношением количества безаварийных прогонов к общему количеству: $X = \frac{40}{50} = 0{,}8$.

Наработка на отказ $X = \frac{|ES'|}{|ES^{\Phi}|}$, где $ES^{\Phi} \subseteq ES$, $\forall YS \in ES^{\Phi} : \exists M_n \in \mathcal{R}(M_0) : M_0 \overset{YS}{\Rightarrow} M_n$, $M_n = M^{\Phi}$, а $ES' \subseteq ES, \forall YS \in ES' : \exists M_n \in \mathcal{R}(M_0) : M_0 \overset{YS}{\Rightarrow} M_n$, $M_n \in \Omega$, т.е. показатель определяется как отношение количества безотказных прогонов к прогонам с отказами: $X = \frac{30}{10} = 3$.

Отметим, что имитация отказов отдельных операций с заданными законами распределения вероятностей происходит по методу Монте Карло программным способом с привлечением генератора случайных чисел и поэтому эту операцию невозможно отобразить аналитически, используя только аппарат окрашенных сетей Петри.

11. Анализ результатов имитационного моделирования.

Результаты проведенного имитационного моделирования представлены в таблице 8. Результаты показывают, что бизнес-транзакция атомарна с вероятностью 0,975. То есть существует вероятность 0,025, что денежные средства клиента, отправленные им через терминал, не будут переведены на его счет. В целом бизнес-транзакция может завершиться аварией с вероятностью 0,1323. Этот показатель выше вероятности безаварийной работы (0,0242) так как существуют операции, приводящие к аварии бизнес-транзакции, но не влияющие на ее успешное завершение.

По результатам анализа чувствительности такими операциями являются отправка клиенту смс-уведомлений о получении оплаты: высока вероятность, что клиент не сможет получить сообщение вовремя, однако на факт оплаты это не повлияет. Поэтому эти процессы влияют только на атомарность бизнес-транзакции, но не влияют на остальные показатели. Этот же вывод делает ПКБТ, проводя исследование чувствительности (см. рис. 7).

Таблица 8 – Показатели надежности бизнес-транзакции

Показатель	**Мин.**	**Макс.**	**Средн.**
Атомарность (в операциях)	0,9566	0,9924	0,9755
Атомарность (в общем)	0,86	1	0,9758
Вероятность избежания аварий	0,25	1	0,8677
Вероятность устранения отказа	0	0	0
Разрешимость отказов	0	0	0
Вероятность безотказной работы	0,86	1	0,9758

Продолжение таблицы 8

Показатель	Мин.	Макс.	Средн.
Вероятность безаварийной работы	0,86	1	0,9758
Частота отказов	0	0,14	0,0242
Частота аварий	0	0,14	0,0242
Наработка на отказ	6,1429	49	36,5595
Наработка на аварию	6,1429	49	36,5595
Интенсивность отказов	0,0204	0,1628	0,0342
Интенсивность аварий	0,0204	0,1628	0,0342

В среднем на каждые 36 успешных платежей происходит один аварийный. Показатели отказов и аварий близки друг к другу, так как в бизнес-транзакции не предусмотрено никаких средств повышения надежности, любой отказ приводит к аварии. По тем же самым причинам показатели разрешимости и устранения отказа равны нулю.

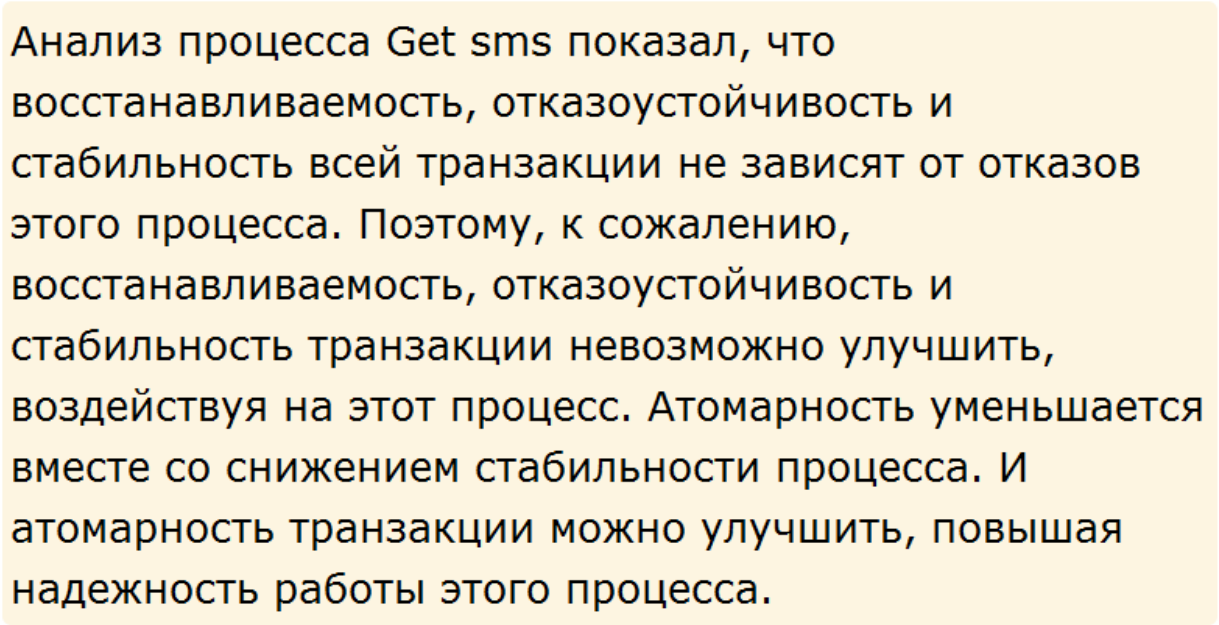


Рисунок 7 – Пример аналитического вывода ПКБТ

Анализ устойчивости с помощью цепей Маркова выявил 15 различных состояний системы. Каждое состояние системы описывается отдельной маркировкой сети Петри. Например, нахождение переменной *data* в позиции «*Pay info*» означает отдельную маркировку, в следующей позиции – другую маркировку. Переход «Get notif» раздваивает ход выполнения бизнес-транзакции, поэтому количество возможных маркировок увеличивается. В целом, анализ показал, что бизнес-транзакция устойчива, однако выполняет свою работу после достижения финальной маркировки. Действительно, после проведения платежа некоторое время работа продолжается – система уведомляет клиента.

3.2. Исследование надежности бизнес-транзакции интернет-магазина

Применим предложенную методику и программный комплекс к проектируемой бизнес-транзакции интернет-магазина, выполняемой на предприятии «ООО Группа Компаний Инвест38» (далее – компания).

На этапе проектирования бизнес-транзакции была поставлена задача обеспечения ее надежности на уровне 90% успешного выполнения. То есть, иными словами, вероятность успеха выполнения должна быть не менее 0,9. Следует учитывать, в этот показатель не включается вероятность того, что товар не будет доступен, и сделка вообще не состоится – наоборот, факт подтверждения недоступности товара может считаться нормальным завершением бизнес-транзакции.

Проведем исследование по алгоритму, предложенному в 2.9.

3.2.1. Идентификация и анализ структуры бизнес-транзакции

Бизнес-транзакция (далее БТ) проводится с целью обеспечения процесса реализации товаров информационного качества, т.е. нематериальной природы. Особенность этого товара в том, что он, как и любая информация, может устаревать и становится неактуальным. Проверка актуальности товара требует ряда ручных операций, проводимых оператором системы. Кроме того, товар может быть временно запрещен к продаже. Информация о его доступности хранится в информационной системе, но в силу особенностей протекающий в компании бизнес-процессов также должна проверяться вручную.

Интернет-магазин, по сути, целиком предназначен для осуществления этой транзакции. Он не является типовым, требует работы не только веб-сайта, но и специально разработанного программного и аппаратного обеспечения, а также использует сложную логику взаимодействия участников. Поэтому необходимо подробно описать работу этой системы.

В процессе задействованы следующие участники:

- Клиент – лицо, выполняющее определенную последовательность действий в БТ с целью приобретения товара.
- Интернет-магазин – комплекс программно-аппаратных средств, предоставляющий клиентам доступ к функциям покупки и оплаты товаров через сеть интернет.
- Интернет-витрина – комплекс программно-аппаратных средств, предоставляющий клиентам возможность рассмотрения и выбора товаров.

- Оператор магазина – сотрудник предприятия, отвечающий за проверку качества товара и обслуживающий процесс покупки.
- Подсистема поддержки работы оператора – программное средство, автоматизирующее деятельность оператора. Оператор взаимодействует с процессом транзакции только через это программное обеспечение.
- Внутренняя информационная система компании – информационная система, обеспечивающая поддержку основных бизнес-процессов компании.
- Система sms-уведомлений – комплекс программно-аппаратных средств, предоставляющий для интернет-магазина возможность уведомления клиента посредством sms-сообщений.
- Платежная система – комплекс программно-аппаратных средств, предоставляющий клиенту возможность оплачивать товар безналичным путем через сеть интернет и предоставляющий информацию о своей деятельности для интернет-магазина.
- Товар – структурированный набор данных, представляющий ценность для клиента. Блок информации формируется в процессе работы интернет-магазина и оператора.

Интернет-витрина и интернет-магазин представляются для клиента в виде веб-сайтов, с которыми он взаимодействует. Оба сайта принадлежат предприятию, однако в силу технических причин физически расположены на различных технологических площадках. Также как и внутренняя информационная система, авторизованный доступ к которой имеют только сотрудники компании. Платежная система и система sms-уведомлений принадлежат другим разным компаниям. Таким образом, в процессе операции осуществляется B2B-взаимодействие между владельцем интернет-магазина, провайдерами платежной системы и системы уведомлений. B2C-взаимодействие проявляется при работе клиента с двумя различными сайтами. Оператор и клиент напрямую не взаимодействуют.

В совокупности представленная система покупки представляет собой распределенную сервис-ориентированную среду взаимодействия: доступ к функциям платежной системы, системы уведомлений, подсистеме оператора, функциям витрины и внутренней ИС осуществляется как к функциям веб-сервисов с помощью соответствующих технологий. Интернет-магазин играет роль оркестратора, управляя основным процессом подготовки объекта к продаже. В целом же процесс покупки является построен как хореография, где ряд участников находится вне зоны ответственности оркестратора, но выполняют процесс для достижения определенной цели. Таким

образом, весь процесс взаимодействия является оркестровкой с элементами хореографии.

Рассмотрим этот бизнес-процесс подробнее. Для схематичного отображения процесса будут использоваться элементы методологии DFD, однако мы не ставим целью соблюдать этот стандарт и воспользуемся лишь отдельными его положениями, так как для понимания сути процесса нет необходимости моделировать потоки данных, хранилища информации и функции ее обработки со всей скрупулезностью. Оркестратор процесса – интернет-магазин на диаграммах не отображен, однако он контролирует большинство описанных функций. Все детали, несущественные для основного процесса, такие как аутентификация и авторизация участников, хранение и подготовка промежуточных данных и др., опускаются. Сложные процессы взаимодействия, включающие несколько этапов, играющих служебную роль (например, согласование параметров протоколов или инициализирующие сообщения), упрощаются до единичных потоков информации.

На рис. 8 отображен процесс покупки товара в окружении источников и приемников информации – клиента, интернет-витрины, подсистемы оператора, платежной системы и системы sms-сообщений. Вне области ответственности бизнес-транзакции находится работа клиента с интернет-витриной: по сути, эта работа пользователя с веб-сайтом, и пользователь может использовать сайт не в целях покупки товара, а, например, для анализа цен и предложения, поэтому бизнес-транзакция покупки активируется после явного решения клиента об этом.

На рис. 9 процесс представляется в виде двух составляющих: инициализации и собственно покупки. Источники и приемники информации остаются такими же. Детально каждый из подпроцессов рассмотрен на рис. 10 и рис. 11. Запрос со стороны клиента с информацией о выбранном товаре инициализирует процесс покупки. Инициализация включает добавление информации о товаре и клиенте в список активных заказов, а также ряд служебных операций. Информация о заказе передается в процедуру проверки доступности товара.

Следует отметить ряд важных особенностей предлагаемого товара. Как уже отмечалось, он представляет собой блок информации, представляющий ценность для клиента. Как любая информация, этот товар обладает способностью к многократной продаже. Однако, как и некоторые виды продаваемой информации (уникальные фотографии, лицензии на программное обеспечение), должен быть продан только однократно, или должна быть запрещена его продажа в течение определенного времени.

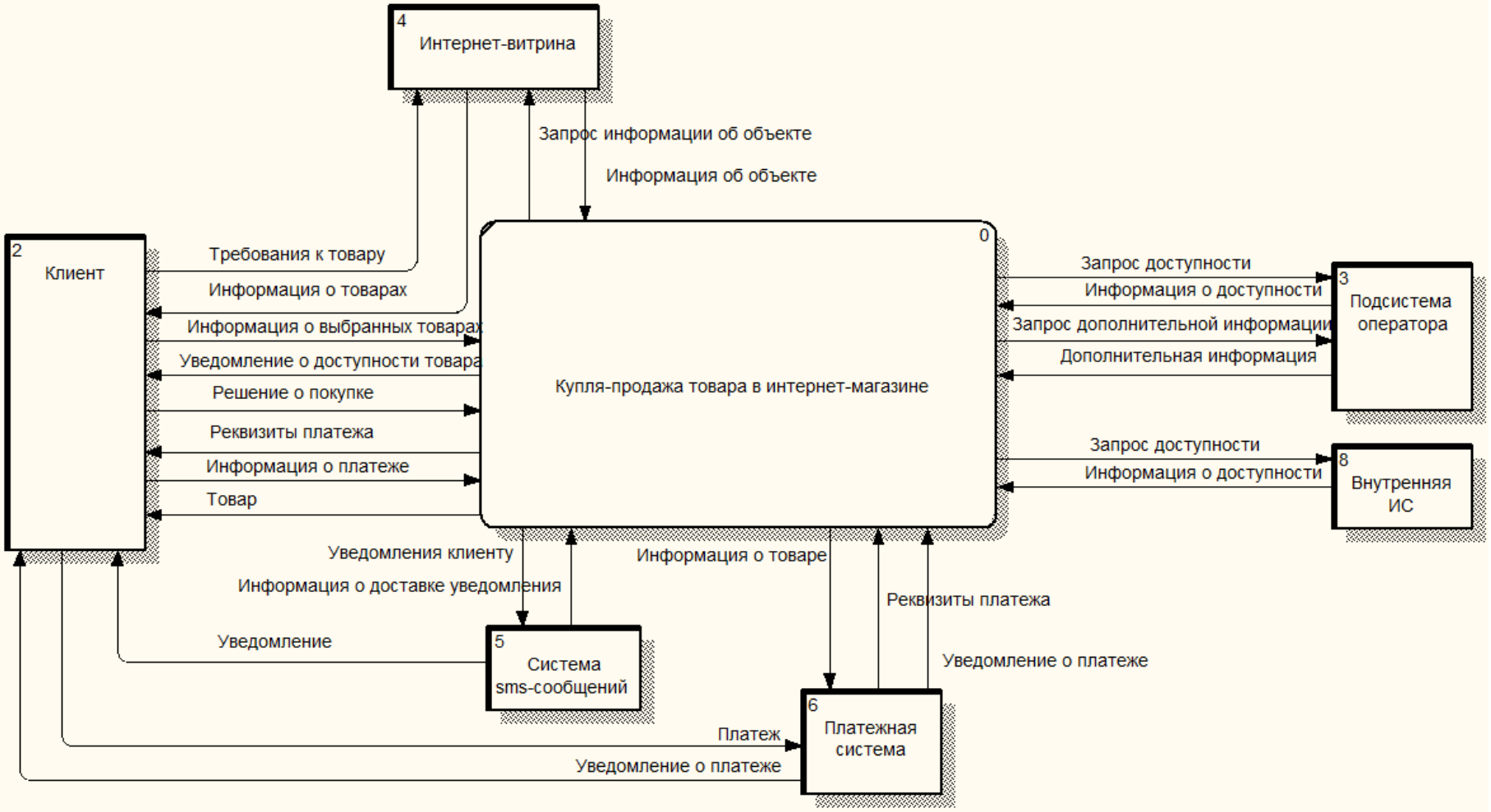


Рисунок 8 – Контекстная диаграмма потоков данных бизнес-транзакции

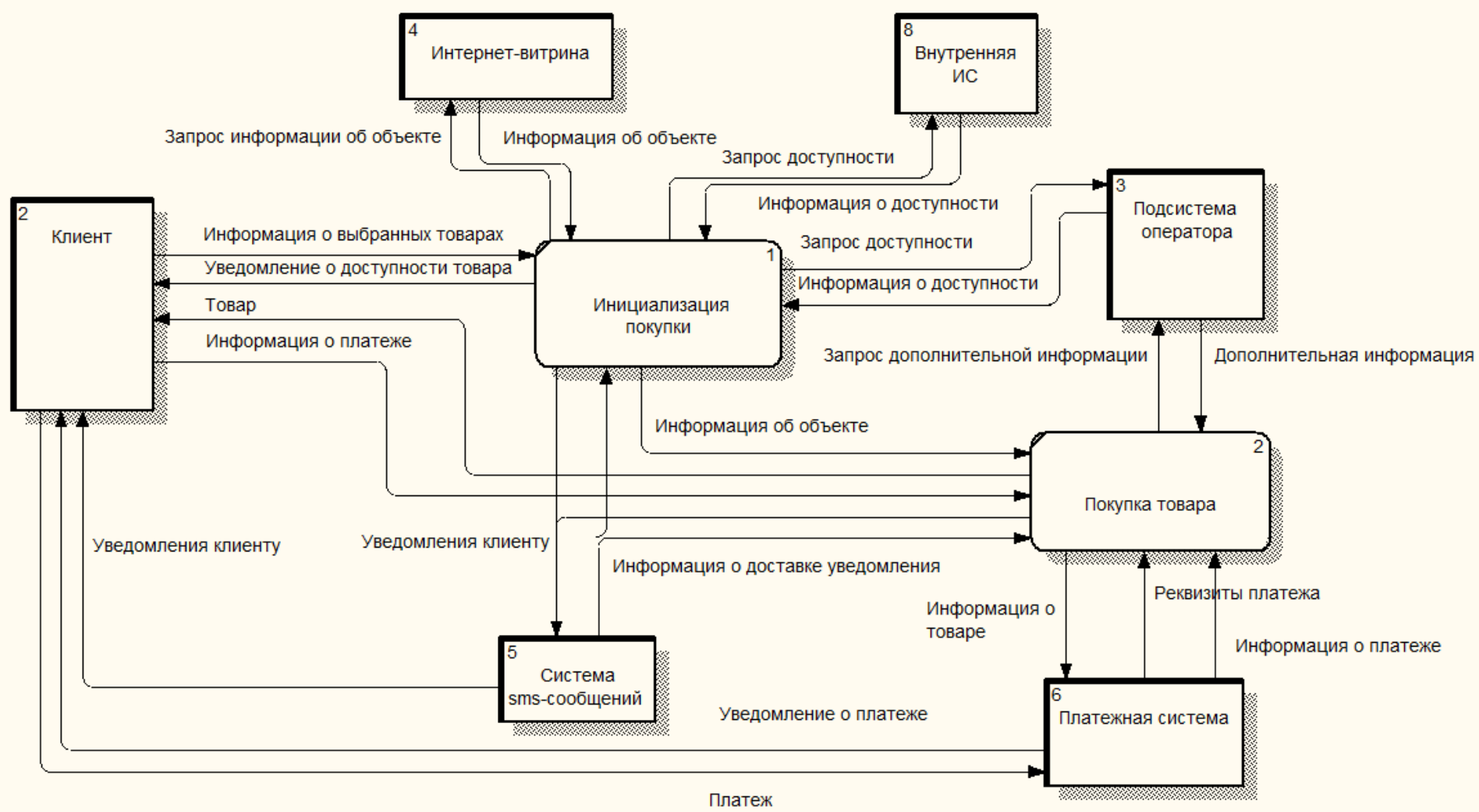


Рисунок 9 – Два основных процесса преобразования данных бизнес-транзакции

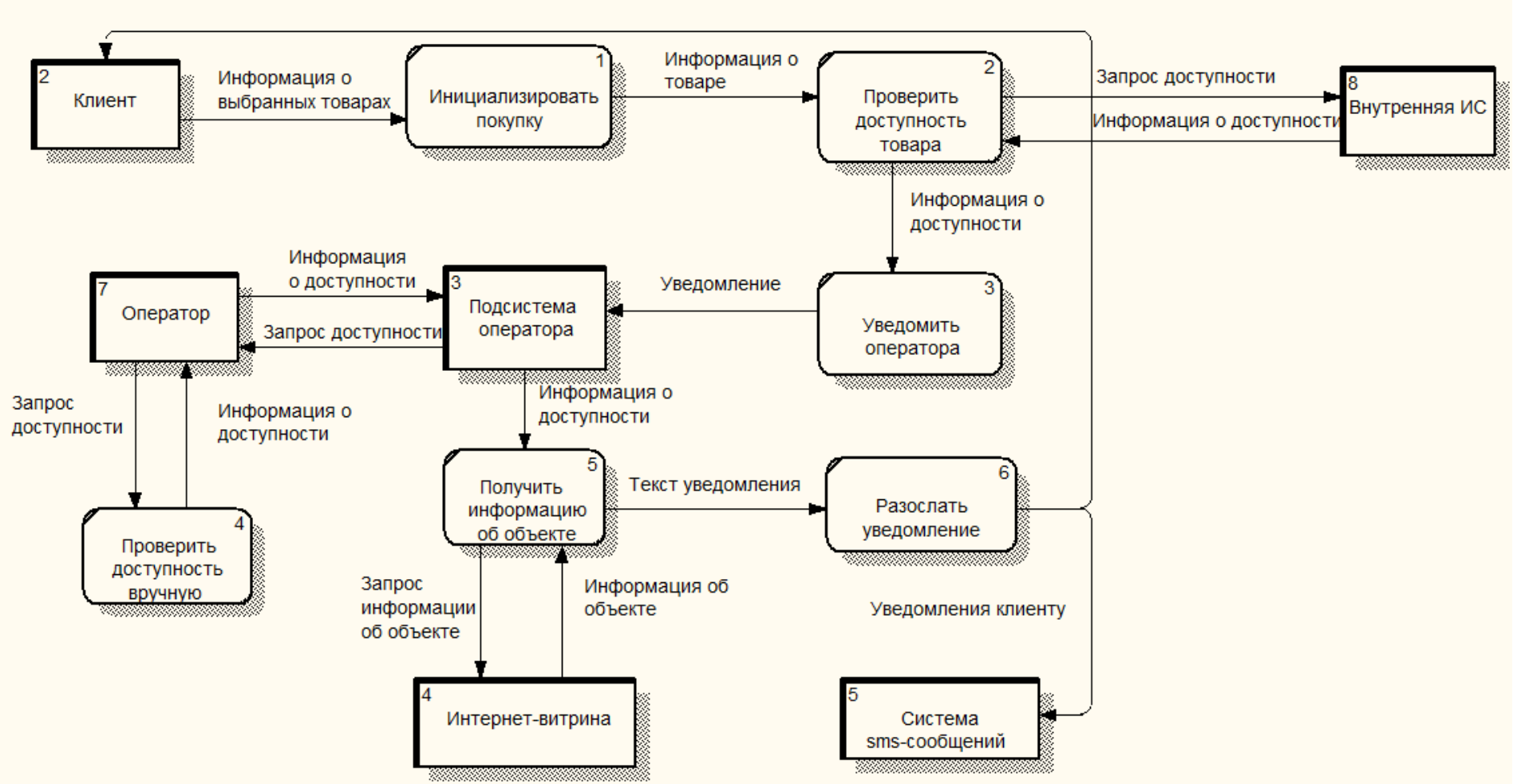


Рисунок 10 – Процесс инициализации покупки

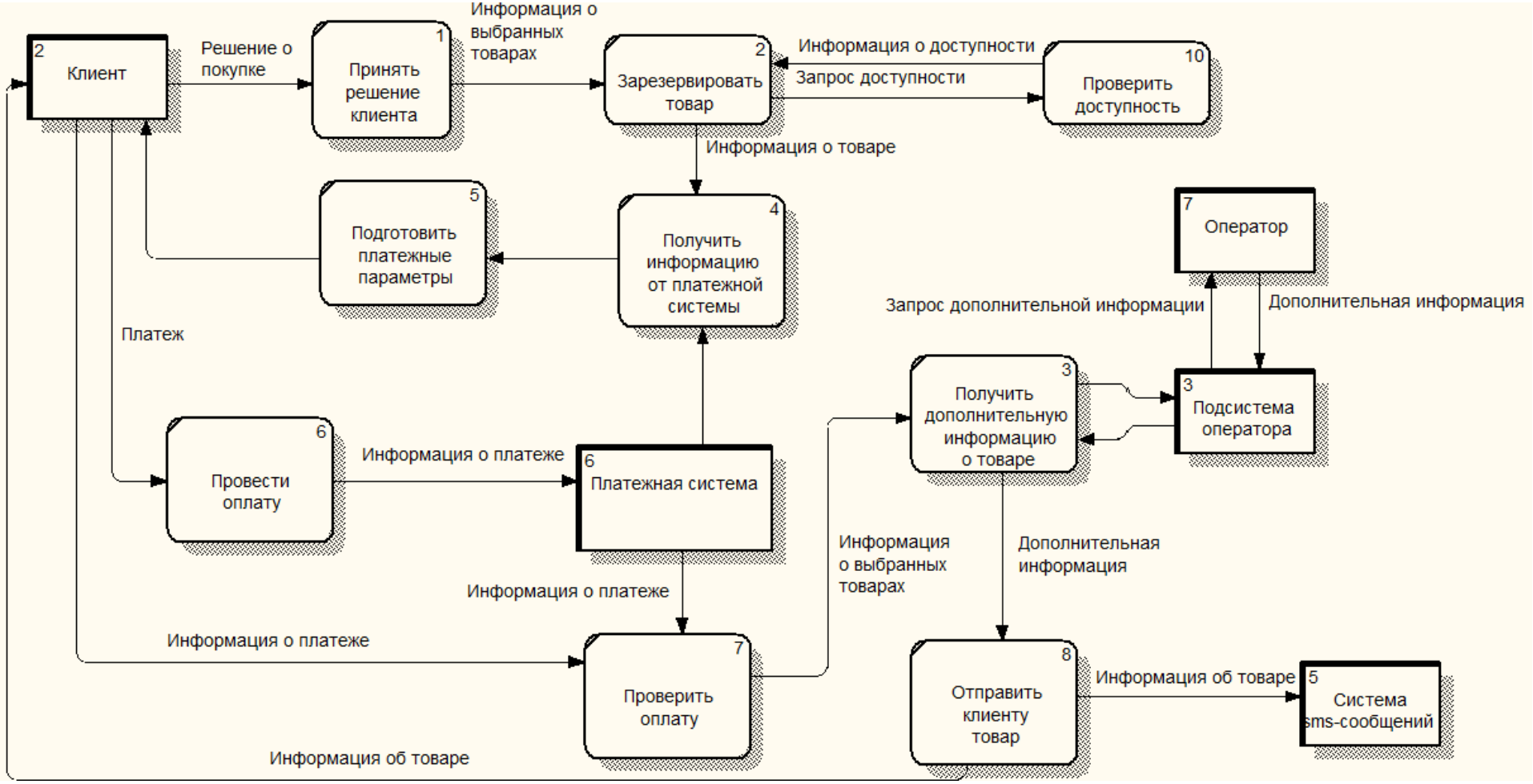


Рисунок 11 – Процесс покупки

Кроме того, периодически товар становится недоступным для продажи. Недоступность продажи в силу специфики товара выясняется в два этапа – сначала через внутреннюю информационную систему, а затем и вручную оператором. Данные во внутренней информационной системой обладают проблемой устаревания: информация о доступности в ней может быть неактуальна, однако в отличие от работы оператора, внутренняя ИС позволяет обрабатывать множество запросов в короткие сроки. Ее база данных постоянно обновляется, но уступает операторской работе в точности и достоверности.

Поэтому на первом этапе, если по данным внутренней информационной системы товар помечен как доступный, то осуществляется запрос в сторону оператора для точного выяснения доступности. Оператор осуществляет ручную проверку доступности и уведомляет о результатах интернет-магазин. Если товар недоступен, то подсистема оператора также изменяет состояние доступности товара для внутренней ИС.

Доступность товара позволяет интернет-магазину продолжить процесс продажи: он запрашивает дополнительную информацию у интернет-витрины и передает уведомление клиенту о возможности покупки. Уведомление содержит также дополнительную информацию, которой клиент мог не располагать. В силу общей ненадежности каналов связи интернет-сайта с клиентом и специфики товара, уведомления осуществляется через два «канала» – посредством коротких сообщений sms и уведомление через «личный кабинет» интернет-магазина. В случае если клиент получает хотя бы одно уведомление, он может продолжить покупку. На этом этапе клиент определяет список товаров, которые он бы хотел приобрести при условии их доступности и появления новой информации о товарах.

В подпроцессе покупки клиент передает интернет-магазину список товаров, которые он решил приобрести. Ввиду того, что клиент может принимать решение длительное время, интернет-магазин вновь проверяет доступность товара, однако для этого достаточно обратиться к специализированной базе данных, так как оператор знает о процессе покупки товара, отслеживает его состояние и, в случае смены его статуса, оперативно самостоятельно меняет запись в базе данных. В случае доступности товар резервируется, т.е. становится недоступным некоторое время для продажи другим клиентам, что гарантирует уникальность владения товаром клиентом некоторое время. Факт

резервирования товара также сказывается на осуществлении других основных процессов компании, не связанных с продажей через интернет. Товар снимается с резерва через определенные периоды времени, за эту операцию рассматриваемая бизнес-транзакция не отвечает.

Успешное резервирование товара позволяет интернет-магазину передать информацию о товаре платежной системе и получить реквизиты платежа. Реквизиты платежа передаются клиенту. Одновременно уведомляется оператор, задачей которого является получение дополнительной информации о товаре, вернее, дополнение информации в товаре. Эта операция проводится вручную, полученная информация хранится в подсистеме оператора, и передается интернет-магазину по специальному запросу.

Клиент оплачивает товар. Информация о платеже поступает и от клиента, и от платежной системы. Интернет-магазин сравнивает хранимую информацию о заказе, информацию о платеже со стороны клиента, информацию от платежной системы и принимает решение о передаче товара клиенту. Факт успешной оплаты отражается в уведомлении клиента. Далее интернет-магазин запрашивает дополнительную информацию от оператора, соединяет ее со всей имеющейся информацией и уведомляет клиента о поступлении товара в его личный кабинет. Этим операция продажи товара завершается.

Во время проектирования бизнес-транзакции все подпроцессы были инкапсулированы в отдельные операции, часть из которых стали операциями, выполняемыми интернет-магазином, а часть вошли в состав веб-служб от других транзакции. Таким образом, абстрагируясь от несущественных технических деталей реализации, можно выделить следующие службы и их операции, участвующие в транзакции (см. табл. 9).

Покажем взаимодействие сторон также с помощью упрощенной диаграммы последовательностей UML, разработанной техническим персоналом компании (см. рис. 12). Из этой схемы видно, что интернет-магазин в разрывах фокуса активности передает управление транзакцией сторонним сервисам и операциям. Чтобы возобновить активность магазина, удаленные службы вызывают его соответствующие сервисные операции.

Таблица 9 – Операции бизнес-транзакции

Название	Вызывается	Вид	Описание
Внутренняя информационная система компании			
Проверка доступности товара	Интернет-магазин	Веб-сервис	Проверяет доступность товара для продажи
Резервирование товара	Интернет-магазин	Веб-сервис	Резервирует товар, предотвращая продажу другим клиентам и участие товара в других процессах компании
Подсистема оператора			
Уведомление оператора	Интернет-магазин	Веб-сервис	Предупреждает оператора о начале операции купли-продажи товара, инициирует ручную проверку доступности
Внесение дополнительной информации	Оператор	Функция настольного приложения	Добавление оператором дополнительной информации по товару
Получение дополнительной информации	Интернет-магазин	Веб-сервис	Возвращает дополнительную информацию, приготовленную оператором
Интернет-витрина			
Получение информации о товаре	Интернет-магазин, подсистема оператора	Веб-сервис	Возвращает информацию о товаре, доступную на витрине
Система «sms»-уведомлений			
Отослать уведомление sms	Интернет-магазин	Веб-сервис	Отсылает готовое sms-сообщение клиенту

Продолжение таблицы 9

Название	Вызывается	Вид	Описание
Платежная система			
Получить информацию от платежной системы	Интернет-магазин	Веб-сервис	Подготавливает платежные параметры на основе данных о клиенте и товаре. Отображает состояние готовности платежной системы.
Осуществить платеж	Клиент	Функция веб-сайта	Осуществляет платеж для клиента
Проверить оплату	Интернет-магазин	Веб-сервис	Возвращает информацию о проведенном клиентом платеже
Интернет-магазин			
Инициализация покупки	Клиент	Функция веб-сайта	Инициализирует процесс купли-продажи
Сообщить о доступности	Подсистема оператора	Веб-сервис	Рассылает уведомления клиенту о доступности товара к продаже
Отослать уведомление в «личный кабинет»	Интернет-магазин	Веб-сервис	Отсылает сообщение клиенту в «личный кабинет»
Принять решение клиента	Клиент	Функция веб-сайта	Принимает информацию от клиента о товаре, который тот решает купить

Продолжение таблицы 9

Название	Вызывается	Вид	Описание
Подготовить платежные параметры	Интернет-магазин	Веб-сервис	Формирует набор платежных инструкций для клиента на основе информации о товаре и информации от платежной системы
Проверить оплату	Клиент	Функция веб-сайта	Проверяет информацию от клиента и платежной системы о проведенном платеже
Уведомить об успешной оплате	Интернет-магазин	Веб-сервис	Уведомляет клиента о том, что его платеж подтвержден
Сообщить итоговую информацию	Интернет-магазин	Веб-сервис	Отсылает клиенту информационный пакет товара.

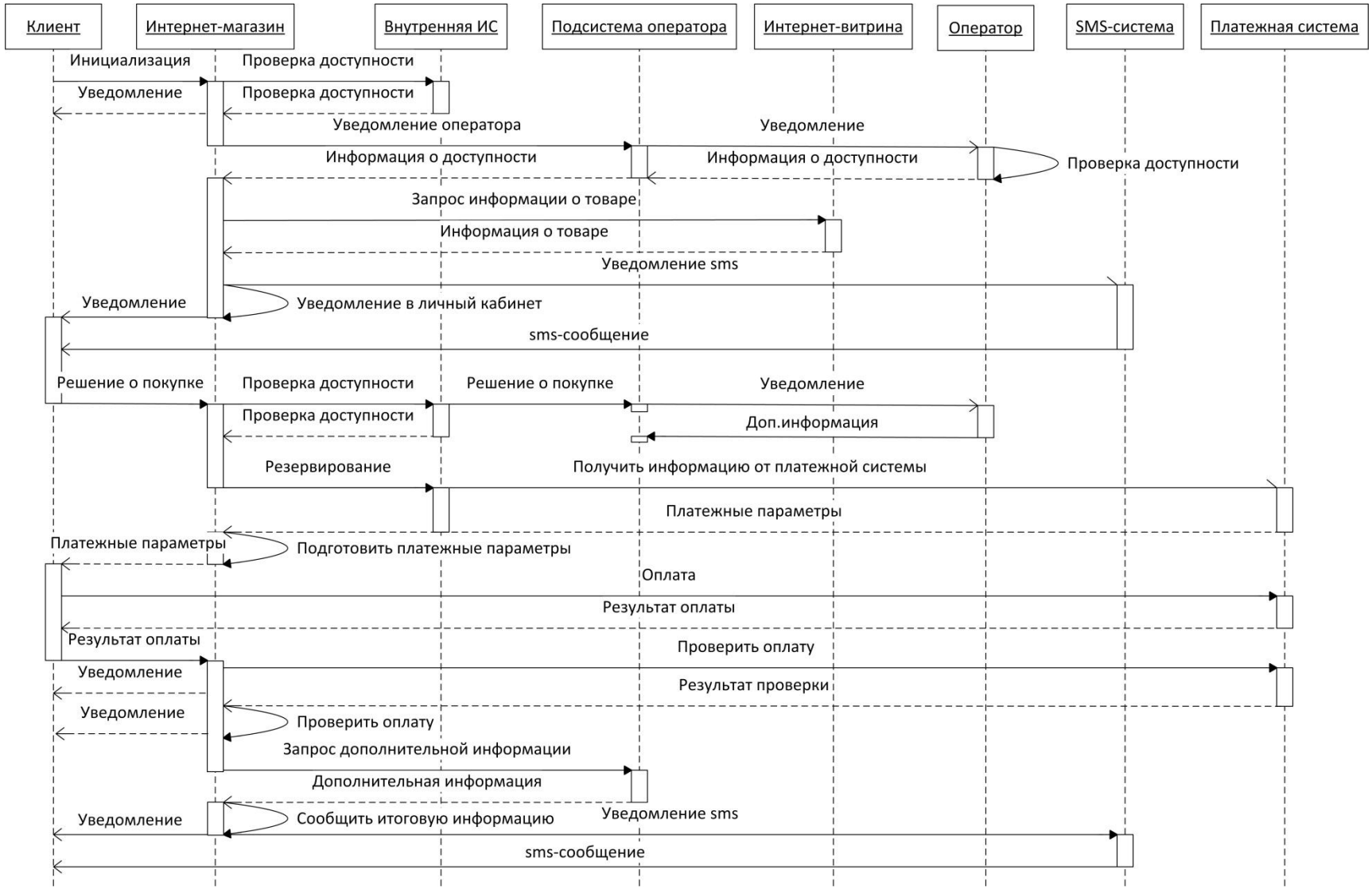


Рисунок 12 – Диаграмма последовательностей бизнес-транзакции

Таким образом, в перерывах между активностью магазина транзакция выполняется по схеме хореографии, а в период активности оркестратором является интернет-магазин. Подобная асинхронность предусматривается в том случае, когда задержки выполнения операции превышают пределы допустимого ожидания. Например, задержки в случае работы оператора или клиента могут длиться долгое время. В том случае, когда функции интернет-магазина вызываются клиентом (через веб-сайт) и предполагается, что выполнение займет длительное время, сайт магазина сразу же возвращает клиенту ответ в виде уведомления. Тем самым реализуется асинхронность работы и с клиентом.

3.2.2. Анализ проблемы оценки надежности исследуемой системы

Согласно природе бизнес-транзакций предполагается, что повышение ее внутренней стабильности можно достичь только изменением ее структуры или введением восстанавливающих последовательностей. Методы, повышающие стабильность интенсивным путем, например, устранением ошибок в программном обеспечении, не входят в пределы ответственности оркестратора бизнес-транзакции, поэтому мы не можем их рассматривать. Таким образом, физически повышение стабильности рассматриваемой бизнес-транзакции возможно только с помощью дублирования процессов (не обязательно аналогами этих процессов, но и целыми последовательностями).

Как мы уже отмечали, моделирование и наглядное представление бизнес-процессов или распределенных взаимодействий осуществляется зачастую в общепринятых нотациях (типа UML и др.), либо по стандартам, принятым на предприятии, осуществляющим проектирование и разработку бизнес-транзакций. В любом случае, оценка надежности с помощью таких схем осуществляется стандартным путем: поэлементная оценка определенных показателей надежности, а затем расчет надежности системы в целом путем анализа ее структуры.

Показатели надежности выбираются экспертным или опытным путем в зависимости от природы элемента, возможностей оценки его характеристик и целей проектирования и расчета надежности системы. Например:

- Вероятность отказов сторонних систем – платежной системы или смс-шлюза определяется их владельцами и закрепляется в численном виде на уровне договорных отношений между интернет-магазином и соответствующими системами.

- Для исследования вероятности отказов внутренних подсистем использовались специальные фоновые программы, так называемые «демоны», тестирующие работоспособность подсистем с заданными интервалами времени. Если во время очередной проверки подсистема не смогла обработать запрос, то он считается отказавшей. Определенную сложность составило изучение отказов работы операторов – специально разработанное программное обеспечение проверяло их готовность в обработке заявки в течение рабочего дня.
- В качестве информации по отказам со стороны клиентов была применена статистика, собранная по результатам работы предыдущей, устаревшей версии интернет-магазина.
- Для сбора информации о доступности товара применялось специальное программное обеспечение, которое в течение определенного интервала времени имитировало поступление заявок на случайные товары. Затем проверялось, является ли данный товар доступным. Проверка, как и при работе интернет-магазина, выполнялась в два шага: запросом к внутреннему серверу компании и ручной проверкой оператора.

3.2.3. Оценка возможностей повышения надежности

Анализ надежности по структурным схемам, как уже нами отмечалось, является хорошо изученной задачей в рамках общей теории надежности. Однако структурный расчет всегда предполагает расчет надежности элементов, однако при проектировании бизнес-транзакции, ввиду ее природы, статистическую оценку работоспособности входящих сервисов получить сложно. Например, сервис может содержать ручные операции, находится вне пределов контроля и ответственности предприятия – проектировщика. Или в нашем примере провайдеры платежной системы и систем sms-уведомлений оценивают лишь один показатель – стабильность в виде вероятности безотказной работы, чего может быть недостаточно для оценки надежности бизнес-транзакции в целом.

Кроме того, крайне важным недостатком использования обычных подходов к проектированию такой динамической системы, как бизнес-транзакция не позволяют проводить проверку на интерфейсное соответствие между операциями и имитационное моделирование процесса ее выполнения. Для успешного проведения имитационного эксперимента с помощью окрашенных сетей Петри проектировщик бизнес-транзакции подходит к ее планированию более тщатель-

но, учитывая возможные нюансы поведения, тестирует ход выполнения и исправляет ошибки. Ошибки, выявленные в процессе проектирования, исправляются дешевле, легче и быстрее чем эти же ошибки, выявленные на более поздних стадиях разработки бизнес-транзакции. Полученная в итоге модель бизнес-транзакции, выдержавшая заданное количество имитационных прогонов гарантирует способность к ее выполнению. Модель, построенная с помощью схематичных рисунков, позволяет только выработать общую концептуальную картину взаимодействия между участниками и операциями транзакции.

Поэтому проведем моделирование бизнес-транзакции этого примера путем описания ее в терминах окрашенных сетей Петри, что позволит определять надежность техникой графического представления и имитационного моделирования. Полное описание бизнес-транзакции дано нами в [234]. Кроме того, показанные ранее схемы работы бизнес-транзакции на первый взгляд кажутся достаточными для ее технической реализации, однако практическая разработка выявит ряд очевидных проблем – например, необходимость рассмотрения вопроса «что будет, если товар окажется недоступным?», который, в свою очередь порождает вопросы вида: «каких участников транзакции затрагивает эта проблема?», «как в этом случае должны вести себя участники?», «как завершить обработку этого товара должным образом?».

Ясно, что обеспечение заданного уровня надежности возможно только с помощью ресурсов, находящихся в пределах ответственности компании, т.к. невозможно влиять на надежность работы клиента или сторонних сервисов. Поэтому необходимо сосредоточиться только на исследовании вклада в общую надежность только со стороны процессов, происходящих внутри компании. В таблице 10 проанализирован характер связанности между участниками для этой бизнес-транзакции, «L» означает слабую связанность, а «S» – сильную.

Так, из таблицы видно, что взаимодействие с любыми внешними процессами для компании в бизнес-транзакции происходит посредство слабой связанности, что открывает ряд преимуществ повышения надежности, связанных с природой сервиса (см. главу 1.1.3): бизнес-транзакция может легко переключаться между сервисами одной функциональности, но разных поставщиков. Таким образом, для повышения надежности этих сервисов можно использовать восстанавливающие сети.

Таблица 10 – Характер связанности участников

	Интернет-магазин	Внутренняя ИС	Подсистема оператора	Интернет-витрина	Оператор
Клиент	L	–	–	L	–
Интернет-магазин	S	–	S	–	–
Внутренняя ИС	S	S	–	–	–
Подсистема оператора	L	–	S	–	L
Интернет-витрина	S	–	–	–	–
Оператор	–	–	L	–	–
SMS-система	L	–	–	–	–
Платежная система	L	–	–	–	–

Однако сильная связанность прослеживается между подсистемами компании, но так как контроль над ними принадлежит ей и есть возможности доработки этих подсистем, их резервирования и внедрения иных способов повышения надежности, то использование восстанавливающих и компенсирующих сетей при проектировании бизнес-транзакции для них также возможно и целесообразно.

3.2.4. Определение целевых показателей надежности

Установим целевые показатели надежности бизнес-транзакции. Как уже отмечалось, частота отказов системы является важнейшим показателем надежности ее работы. Так как бизнес-транзакции по условиям проектирования должна обеспечивать 90% надежность, то установим целевым показателем надежность вероятность безотказной работы на уровне 0,9 (без учета недоступности товара и действий клиента), таким образом, в случае доступности товара, согласия клиента на его покупку и его действий по оплате, бизнес-транзакция должна завершаться успешно с вероятностью 0,9. Важным является показатель атомарности бизнес-транзакции, который говорит о том, что об успешном или неуспешном завершении бизнес-транзакции известно все участникам, т.е. она завершилась предсказуемо для них. Наиболее приемлемым значением атомарности является 1, однако, как было показано в 2.5, это требует гарантированной абсолютной надежности всех входящих элементов или их компенсации вплоть до определенного уровня (тогда и атомарность будет до определенного уровня).

Поэтому зададим пороговое значение атомарности на уровне 0,95. То есть все возникающие отказы бизнес-транзакции должны контролироваться и отслеживаться, и случаев, когда отказ приводит к аварии, должно быть не более 5% в общем количестве запусков.

Проведем исследование всех подпроцессов, которые входят в зону ответственности компании – владельца интернет-магазина (то есть только на их надежность потенциально может влиять компания). Для оценки статистики отказов было проведено раздельное тестирование этих процессов. Отказ фиксировался в случае отсутствия ответа в течение 30 секунд. Для ручного процесса проверки доступности оператором отказ фиксировался при отсутствии реакции в течение 1 часа, заказы, принятые в нерабочее время должны обслуживаться в течение 1 часа первого же периода рабочего времени. Таким образом, была составлена следующая таблица (см. табл. 11).

Таблица 11 – Исследуемые процессы

Название	**Описание**	**Вероятность отказа**
Проверка доступности товара	Проверяет доступность товара для продажи	0,97990
Резервирование товара	Резервирует товар, предотвращая продажу другим клиентам и участие - товара в других процессах компании	0,99376
Уведомление оператора	Предупреждает оператора о начале операции купли-продажи товара	0,96125
Ручная проверка доступности	Оператор проводит проверку доступности товара вручную	0,77852
Внесение дополнительной информации	Добавление оператором дополнительной информации по товару	0,86148
Получение дополнительной информации	Возвращает дополнительную информацию, приготовленную оператором	0,99439
Получение информации о товаре	Возвращает информацию о товаре, доступную на витрине	0,99013
Инициализация покупки	Инициализирует процесс купли-продажи	0,97797
Сообщить о доступности	Рассылает уведомления клиенту о доступности товара к продаже	0,88452
Уведомить об успешной оплате	Уведомляет клиента о том, что его платеж подтвержден	0,88875
Сообщить итоговую информацию	Отсылает клиенту информационный пакет товара.	0,87913

Низкая надежность работы оператора продиктована влиянием особенностей должностных обязанностей оператора и человеческого фактора: выполнение работ для интернет-магазина является второстепенной задачей оператора, поэтому, к сожалению, он не всегда находит время для успешного выполнения этой работы. Кроме того, в этих процессах участвует подсистема оператора, связь с которой от внутренней информационной системы может отсутствовать.

Более высокий уровень отказов процессов отправки данных пользователю можно объяснить следующими причинами: во-первых, отправка данных зависит от сторонних приложений, обеспечивающих сетевую коммуникацию и работу с базой данных. Редко, но эти приложения могут быть недоступны. Во-вторых, отправка данных требует определенного времени, однако настройки серверов провайдера, где находятся все информационные ресурсы компании (витрина, магазин, внутренняя система), позволяют только 15 секундное выполнение одной программы. Таким образом, если программа не успела закончить работу по отправке данных в течение 15 секунд, то она принудительно завершается.

3.2.5. Конструирование модели бизнес-транзакции

Учитывая данное ранее описание сервисов, их операций и транзакций в терминах окрашенных сетей Петри отобразим эту бизнес-транзакцию графически с помощью разработанного ПКБТ (см. рис. 13, 14 и 15, сеть на них изображена последовательно слева направо).

Множество операций и множество участников бизнес-транзакции было определено в таблице 12. Участники представлены различной цветовой гаммой операций.

Позиции «Notific1[1]», «Notific2», «Notific3» имеют инициализирующие строковые значения. Они нужны для автоматического генерирования информации по товару во время выполнения транзакции. Позиция «Buyer wish» содержит список товаров, которые хотел бы приобрести клиент вместе с номером клиента. Для хранения этой информации позиция обладает множеством цветов «b_order».

[1] Названия процессов даны на английском языке ввиду того, что графическая среда не поддерживает международные кодировки символов.

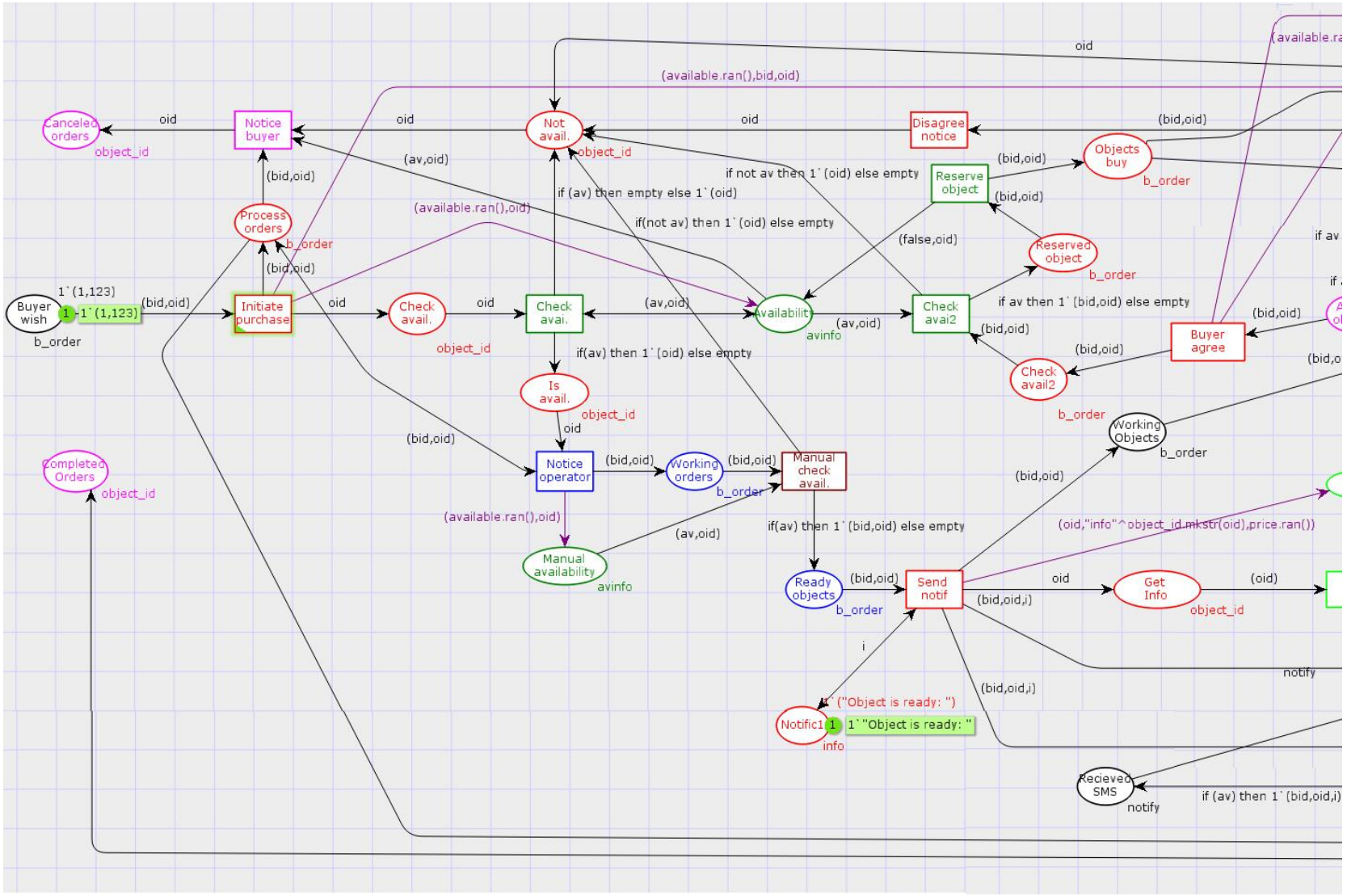


Рисунок 13 – Бизнес-транзакция, описанная с помощью окрашенной сети Петри, часть 1

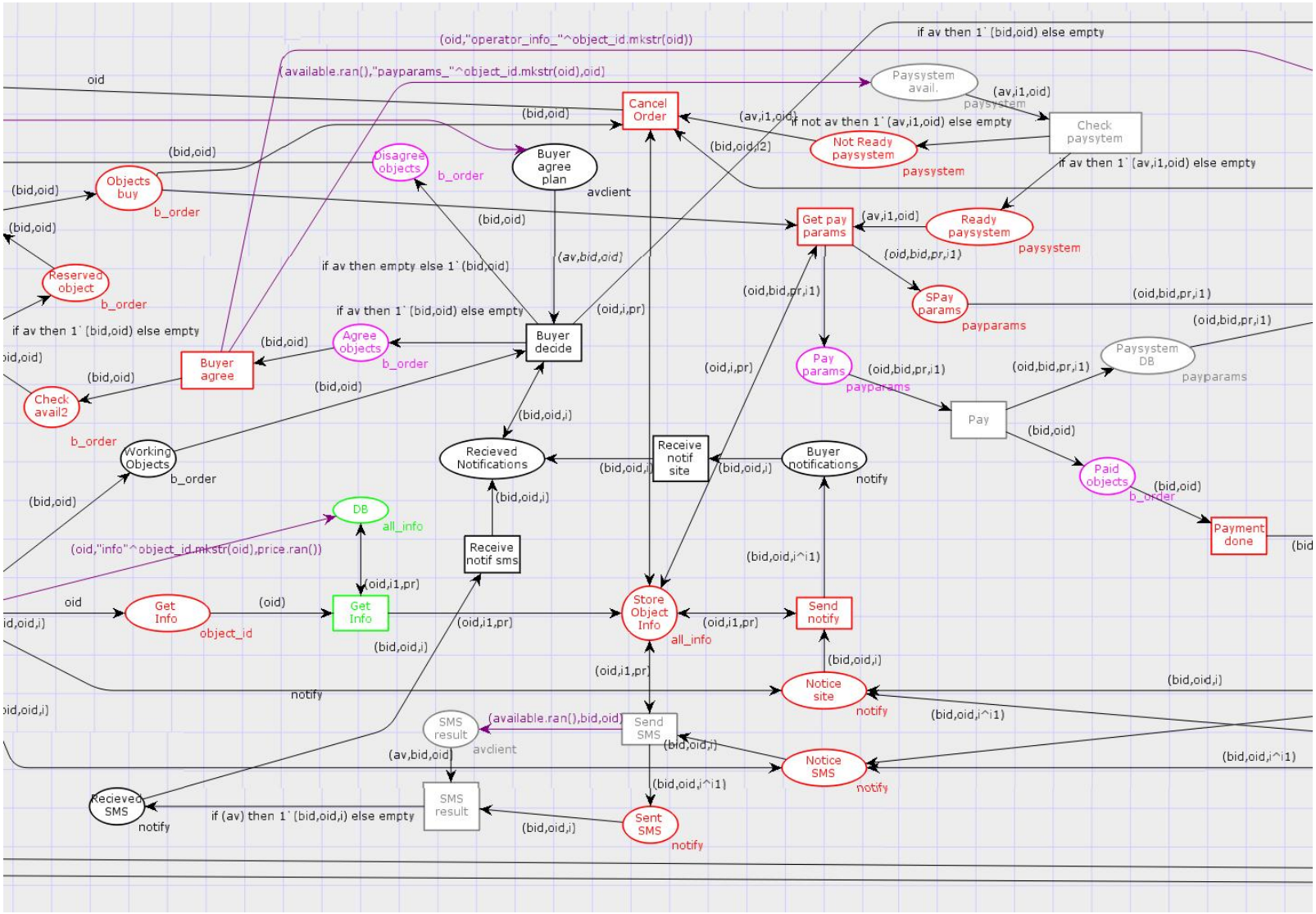


Рисунок 14 – Бизнес-транзакция, описанная с помощью окрашенной сети Петри, часть 2

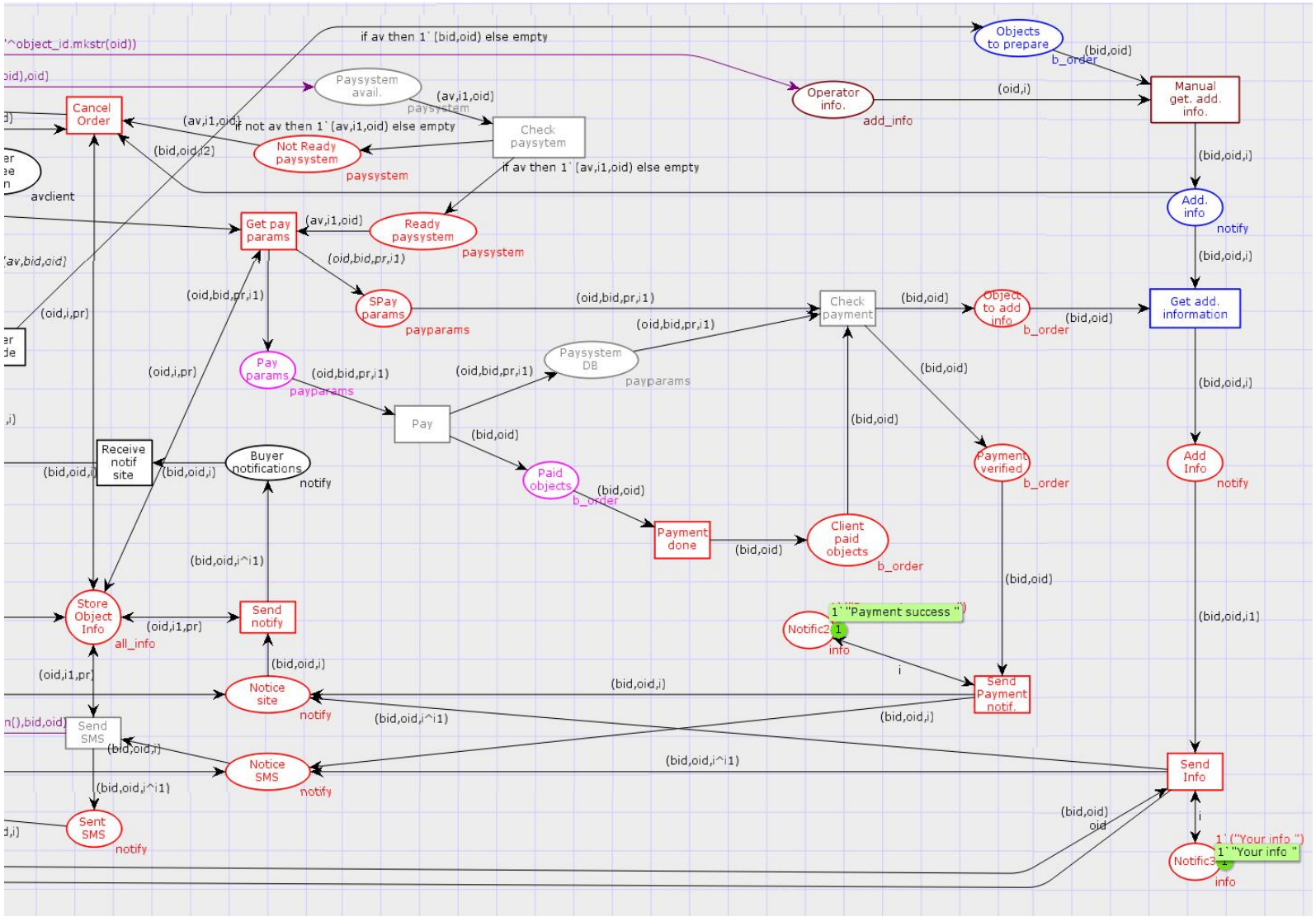


Рисунок 15 – Бизнес-транзакция, описанная с помощью окрашенной сети Петри, часть 3

Позиции «Notific1», «Notific2», «Notific3», «Buyer wish» задают начальную маркировку транзакции. В целях моделирования параллельной работы над несколькими заказами нами были введены позиции BUYERS и OBJECTS, которые поставляют уникальные пары заказа: «номер покупателя», «номер товара».

Множество финальных маркировок определяется метками на позициях «Completed Orders», «Canceled orders», «Recieved Notifications», которые означают список купленных товаров, список некупленных товаров и список полученных клиентом уведомлений. Позиция «Recieved Notifications» является позицией хранения хореографии.

При проведении имитационного эксперимента существующие метки на любых других позициях при том условии, что текущая маркировка мертва, означают ошибки проектирования транзакции. Иными словами, транзакция считает успешно завершенной только в том случае, когда маркировки позиций «Completed Orders», «Canceled orders» являются непустыми, в позиции «Recieved Notifications» находится любое количество меток, а все остальные – пусты. Позиции «Availability», «Manual availability», «DB», «Paysystem avail.», «Operator info.» хранят информацию о доступности во внутренней ИС, информацию о доступности от оператора, информацию о товаре, платежные параметры и дополнительную информацию от оператора и должны заполняться вне пределов ответственности бизнес-транзакции.

В целях успешной имитации их метки генерируются случайным образом с помощью соответствующих выражений над дугами.

Изучение работы оператора и существующей системы автоматизированной обработки заказов позволили получить информацию о том, что на этапе первичного запроса доступности отсеивается в среднем 11% запрашиваемых товаров, а работа оператора позволяет отсеять еще 7%. Таким образом, присутствует вероятность (0,89 * 0,93)=0,8277, что товар будет доступен для продажи после двух проверок. В сети Петри нашего примера доступность товара возвращает булевая функция «avran(param)», параметром «param» которой является соответствующая вероятность доступности. Функция реализует механизм генерирования случайной величины распределения Бернулли.

Кроме того, в нашей задаче важную роль играют процессы доставки sms-сообщений и работоспособности платежной системы, на схеме показана сложная логика работы по проверке этих процессов, а эти системы находятся вне зоны ответственности исследуемой бизнес-транзакции. Таким образом, изучение вероятностей отказов доставки или платежной системы с помощью ПКБТ неце-

лесообразно. От компаний, поддерживающих sms-сервис и платежную систему, были получены вероятности успешной доставки sms и доступности платежной системы – 0,95 (95% sms доставляются до адресата в среднем) и 0,99 (т.к. «up-time» платежного сайта – 99% времени). Эти показатели были использованы на схеме в функции «avran», которая моделирует отказы этих операций.

Еще один стохастический процесс, влияющий на выполнение бизнес-транзакции, это определенность, желание клиента в покупке товара. Часто случается, что клиент инициирует бизнес-транзакцию, но когда для покупки все приготовлено, то он не решается тратить на товар денежные средства. По статистике, это случается в 30% случаев, то есть практически в каждом третьем случае клиент отказывается от текущей покупки и может инициировать покупку другого товара. Случайность решения клиента также зададим с помощью функции «avran».

В процессе разработки и тестирования схемы были замечены проблемы, связанные с вопросами операционных отказов: в случае недоступности товара, несогласия клиента приобрести товар, невозможности обработки платежа платежной системой. Соответствующие позиции участников, содержание информацию об этом товаре, оставались неизменными, что сигнализировало о нарушении хода выполнения: информация о сбое не аккумулировалась ответственными участниками, а оставалась где-то на промежуточном этапе. Эти проблемы выявили необходимость введения компенсирующей последовательности, которая отменяет выполнение того или иного заказа и уведомляет об этом всех участников. Эта компенсирующая сеть состоит из переходов «Disagree notice», «Notice buyer» и позиций «Disagree objects», «Not avail.», «Canceled orders» и запускается, если товар недоступен, клиент отказался покупать, платежная система не в данный момент доступна. Финальной позицией компенсирующей сети является позиция «Canceled orders». Эта же компенсирующая сеть может использоваться для отмены бизнес-транзакции при отказах отдельных операций.

Операция инициализации покупки «Initiate purchase» вызывается клиентом с передачей номера клиента и желаемого товара. Эта информация сохраняется в позиции «Process orders». Операция «Check avai.» проверяет доступность товара. В случае доступности следует переход к уведомлению оператора. Случай недоступности рассматривается как организационный отказ, который вызывает переход к компенсирующей сети. После уведомления оператора сервисом «Notice operator» происходит ручная проверка доступности. Отрицательный результат ручной проверки также может привести к откату транзакции. Уведом-

ление об успешной проверке происходит с помощью вызова операции интернет-магазина «Send notif» после чего он запрашивает информацию у интернет-витрины с помощью функции «Get Info» и посылает уведомление клиенту и так далее. Приведем таблицу соответствия сервисам и операциям рассматриваемой бизнес-транзакции и позиций построенной сети Петри (см. табл. 12).

Таблица 12 – Соответствие операций бизнес-транзакций и переходов сети Петри

Операция, функция	**Переход**
Внутренняя информационная система компании	
Проверка доступности товара	Check avai., Check avai2
Резервирование товара	Reserve object
Подсистема оператора	
Уведомление оператора	Notice operator
Внесение дополнительной информации	Manual get. add. info.
Получение дополнительной информации	Get add. information
Интернет-витрина	
Получение информации о товаре	Get Info
Система «sms»-уведомлений	
Отослать уведомление sms	Send SMS
Платежная система	
Получить информацию от платежной системы	Check paysytem
Осуществить платеж	Pay
Проверить оплату	Check payment
Интернет-магазин	
Инициализация покупки	Initiate purchase
Сообщить о доступности	Send notif
Отослать уведомление в «личный кабинет»	Send notify
Принять решение клиента	Buyer agree
Подготовить платежные параметры	Get pay params
Проверить оплату	Payment done
Уведомить об успешной оплате	Send Payment notif.
Сообщить итоговую информацию	Send Info

Существует несколько позиций, которые были введены намеренно для отражения работы участников, выполняемой вне пределов ответственности транзакции (см. табл. 13). Эти работы не были показаны на диаграмме UML, однако

они могут оказывать существенное влияние на ход выполнения транзакции, а значит и на ее надежность.

Таблица 13 – Дополнительные переходы сети Петри

Позиция	Назначение
Manual check avail.	Имитация ручной проверки доступности оператором
Receive notif site	Имитация получения клиентом уведомления с помощью сайта
Receive notif sms	Имитация получения клиентом уведомления с помощью sms
SMS result	Имитация доставки сообщения sms клиенту
Buyer decide	Имитация процесса принятия решения о покупки клиентом

Построенная модель полностью соответствует заявленным ограничениям на бизнес-транзакцию в сервис-ориентированной среде:

- Определено конечное непустое множество участников бизнес-транзакции.
- Каждая позиция или переход принадлежит какому-либо участнику
- Каждая позиция выражает операцию, которая вызывается только одним участником.
- Для поддержки работы по шаблону хореографии используются позиции хранения.
- Определены начальная и множество финальных маркировок сети. Финальная маркировка предполагает наличие меток только в позициях хранения.
- В бизнес-транзакции рассмотрены все случаи наступления операционных отказов.
- Атомарность транзакции при наступлении операционного отказа поддерживается компенсирующими сетями.

Построенная модель также соответствует схемам UML и DFD, предложенным техническими специалистами компании. Однако в отличие от этих схем модель задает интерфейсные ограничения для операций сети, добавляет информацию о хранимых переменных и данных, и, самое главное, посредством синтаксической проверки и имитационных экспериментов силами ПКБТ доказывается, что эта модель адекватно отражает проектируемую бизнес-транзакцию, и построенное на базе этой модели взаимодействие будет правильно исполняться.

3.2.6. Исследование надежности с помощью ПКБТ

Проведем анализ надежности описанной бизнес-транзакции интернет-магазина с помощью ПКБТ. Первичным прогоном в среде CPN Tools было установлено, что приблизительным значением максимального количества переходов для 50 запусков транзакции в рамках одного эксперимента является значение 2000 переходов (см. рис. 16).

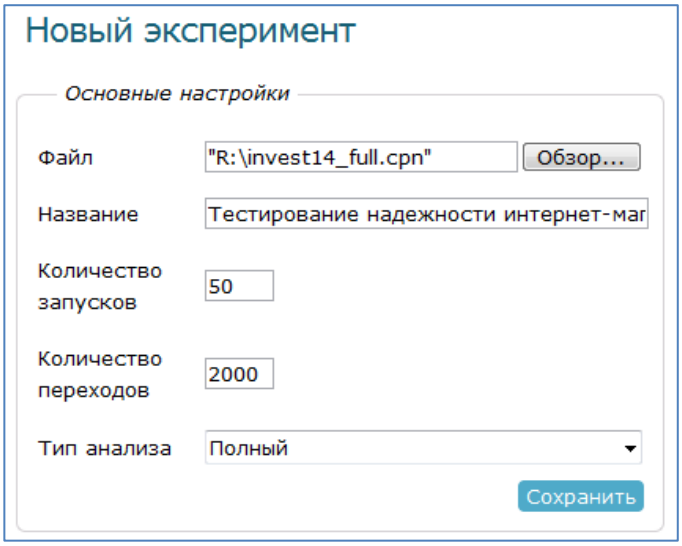


Рисунок 16 – Инициализация эксперимента

В схеме только одна сеть, а успешное выполнение бизнес-транзакции завершается на позиции «Completed Orders». Если выполнение бизнес-транзакции завершается на позиции «Canceled Orders», то имеет место срабатывание компенсирующей последовательности.

Начнем исследование надежности без учета доступности объектов и нерешительности клиента, т.е. минимизируем влияние на бизнес-транзакцию внешних факторов: вероятность доступности товара, решения о покупке клиента, надежность внешних служб будет 1. Таким образом, это позволит отделить бизнес-транзакцию от внешней среды и исследовать ее внутреннюю надежность (см. главу 2.5).

Создадим новый эксперимент в ПКБТ, который позволит исследовать надежность по существующим параметрам системы (см. рис. 17).

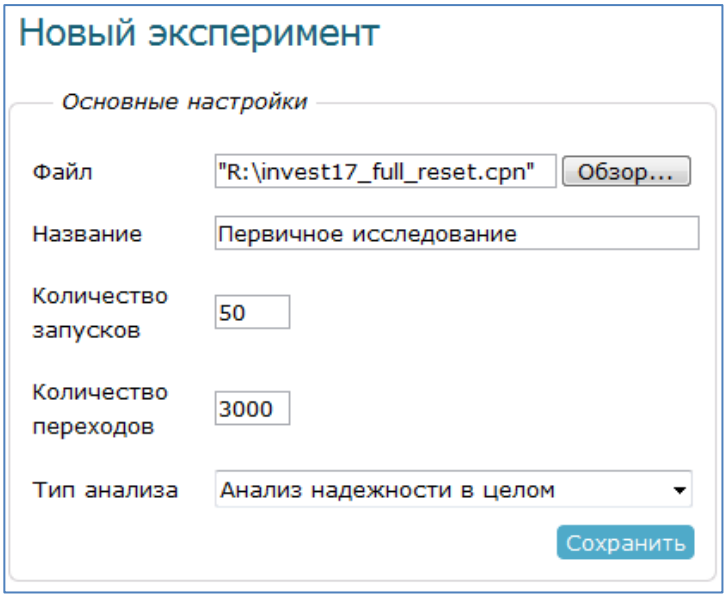


Рисунок 17 – Первичный анализ надежности

В настройках эксперимента зададим все операции (см. рис. 18), которые подвергаются исследованию, в соответствии с таблицей 11. Для начала надежность бизнес-транзакции без учета возможностей компенсации и восстановления.

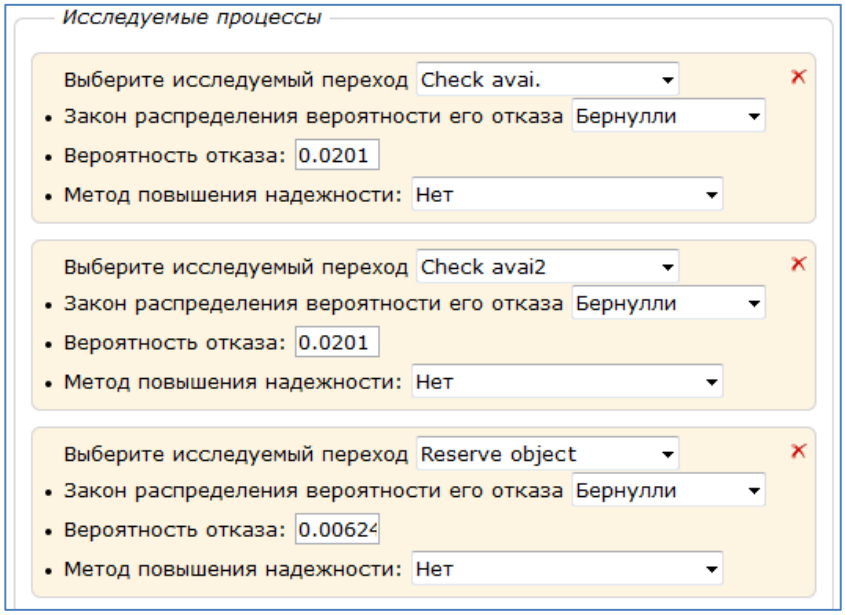


Рисунок 18 – Фрагмент настроек эксперимента

В результате было выяснено, что бизнес-транзакция может успешно выполняться только в 12% случаев (см. табл. 14). Так как любой отказ любой операции бизнес-транзакции приводит к аварии, то и уровни аварийности и отказов одинаковые. Атомарность выполнения отдельных операций составляет 93%, а атомарность транзакции в общем – всего лишь 44%.

То есть в среднем в 56 запусках из 100 бизнес-транзакция не достигает ожидаемого результата – положительного и отрицательного. Стороны, выполняющие бизнес-транзакцию, не уведомляются о результатах ее завершения, что может привести к труднопредсказуемым последствиям.

Атомарность может быть повышена с помощью введения компенсирующих последовательностей, которые позволят «откатывать» ее и предупреждать участников о преждевременном завершении.

Таблица 14 – Результаты первого раунда эксперимента

Показатель	**Значение**
Атомарность (в операциях)	0,07
Атомарность (в общем)	0,44
Вероятность избежания аварий	0,12
Вероятность устранения отказов	0,12
Разрешимость отказов	0,00
Вероятность безотказной работы	0,44
Вероятность безаварийной работы	0,44
Частота отказов	0,56
Частота аварий	0,56
Наработка на отказ	0,82
Наработка на аварию	0,82
Интенсивность отказов	1,32
Интенсивность аварий	1,32

Поэтому подключим компенсирующие сети (см. рис. 19) и проведем второй раунд эксперимента.

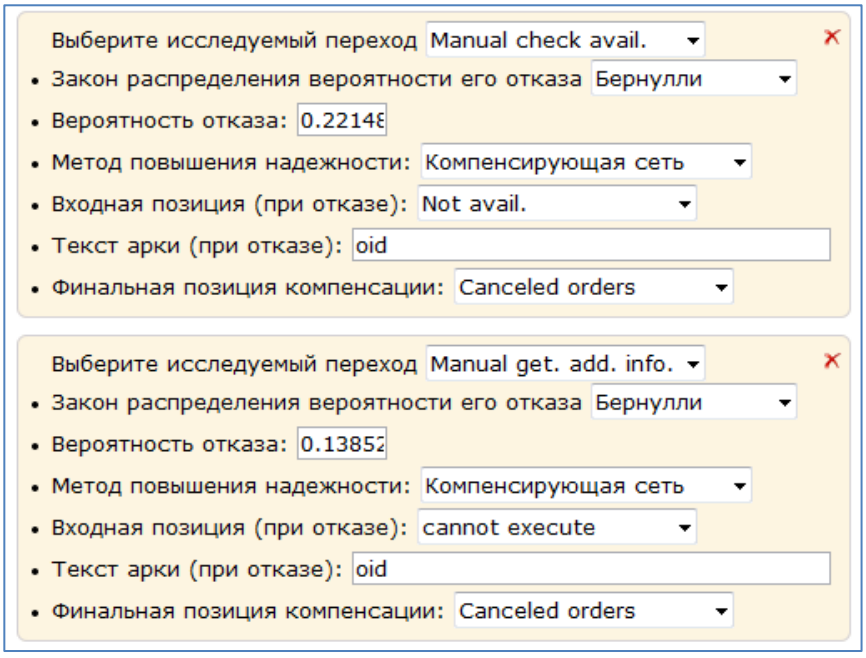


Рисунок 19 – Фрагмент настроек компенсации

Для процессов, где происходит проверка доступности товара, компенсирующая сеть будет начинаться с позиции «Not avail.», что означает недоступность товара, а для остальных – «Cannot execute». Компенсирующие последовательности завершаются единой позицией «Canceled orders». Технически компенсирующая последовательность может быть представлена сервисом или группой сервисов, запуск которые инициирует оркестратор, когда определяет, что произошел сбой выполнения операции. Для операций, которые не влияют на основной ход транзакции, компенсации, естественно, быть не может.

Результаты второго раунда эксперимента показывают, что поведение бизнес-транзакции с учетом компенсации существенно улучшилось (см. рис. 20 и табл. 15).

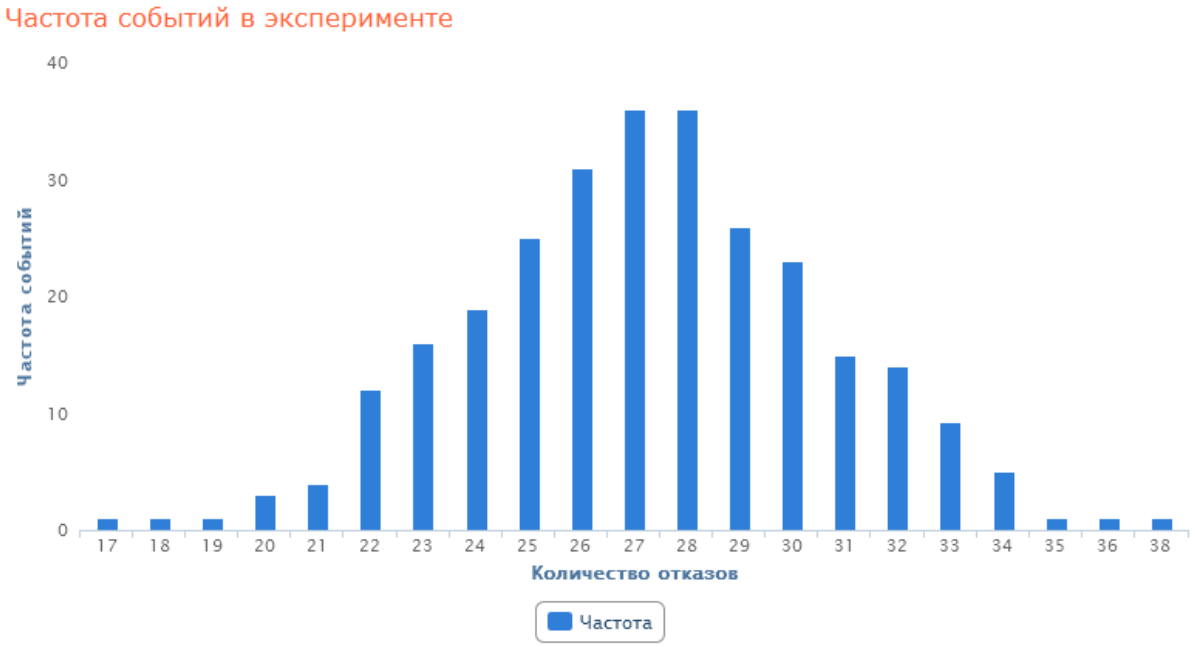


Рисунок 20 – Результаты второго раунда эксперимента (количество отказов на 50 запусков)

Положения и выводы, на которых основывается ПКБТ при работе со статистическими данными, причины использования отдельных методов, информация по технологии планирования экспериментов описаны в Приложении А.

Во-первых, атомарность в целом достигла 91%, то есть в 91% случаев бизнес-транзакция завершается таким образом, что все участники уведомляются об этом, чем поддерживается единство их совместного состояния. Во-вторых, вероятность избежания аварий возросла до 0,88, то есть 88% запусков бизнес-транзакции завершаются определенным, положительным или отрицательным, результатом.

Таблица 15 – Результаты второго раунда эксперимента

Показатель	Значение
Атомарность (в операциях)	0,02
Атомарность (в общем)	0,91
Вероятность избежания аварий	0,88
Вероятность устранения отказов	0,24
Разрешимость отказов	0,78
Вероятность безотказной работы	0,44
Вероятность безаварийной работы	0,91
Частота отказов	0,56
Частота аварий	0,09
Наработка на отказ	0,81
Наработка на аварию	14,25
Интенсивность отказов	1,35
Интенсивность аварий	0,10

Но успешный результат выполнения возможен только в 44% запусков, все остальные случае заканчиваются отменой выполнения или аварией. Проведем анализ надежности отдельных процессов: сравним вероятность их безошибочной работы (см. табл. 16). Как можно отметить, в целом полученные значения в ходе эксперимента соответствуют заданным в условиях (сравн. с табл. 11).

Таблица 16 – Вероятность безошибочной работы процессов

Показатель	Значение
Initiate purchase	0,95
Check avai.	0,98
Get Info	0,99
Notice operator	0,96
Manual check avail.	0,69
Send notif	0,88
Reserve object	0,99
Send Payment notif.	0,88
Get add. information	0,99
Send Info	0,87
Manual get. add. info.	0,85
Check avai2	0,98

Обратим внимание, что процессы «Manual check avail.», «Send Info», «Manual get. add. info.», «Send Payment notif.», «Send notif» имеют существенно более высокую вероятность отказа, нежели чем остальные[1]. Возможно, именно из-за их отказов и отказывает бизнес-транзакция целиком. Для того чтобы в

[1] Известно из схемы процесса, что процессы «Send notif» и «Send Payment notif.» проводят только уведомление клиента и не влияют на результат работы все системы, поэтому мы можем исключить эти процессы из дальнейшего анализа.

этом убедиться, в третьем раунде эксперимента проведем анализ чувствительности надежности бизнес-транзакции к надежности отдельных процессов.

Результаты третьего раунда показали, что почти все исследуемые процессы влияют на стабильность работы бизнес-транзакции. При повышении частоты отказов процесса вероятность отказа всей бизнес-транзакции прямо пропорционально увеличивается. Корреляционное поле, продемонстрированное на рис. 21 для процесса «Initiate purchase», схоже с графиками для других процессов. Такое поведение объясняется тем, что почти все процессы лежат «на критическом пути», т.е. являются обязательными к выполнению в течение бизнес-транзакции.

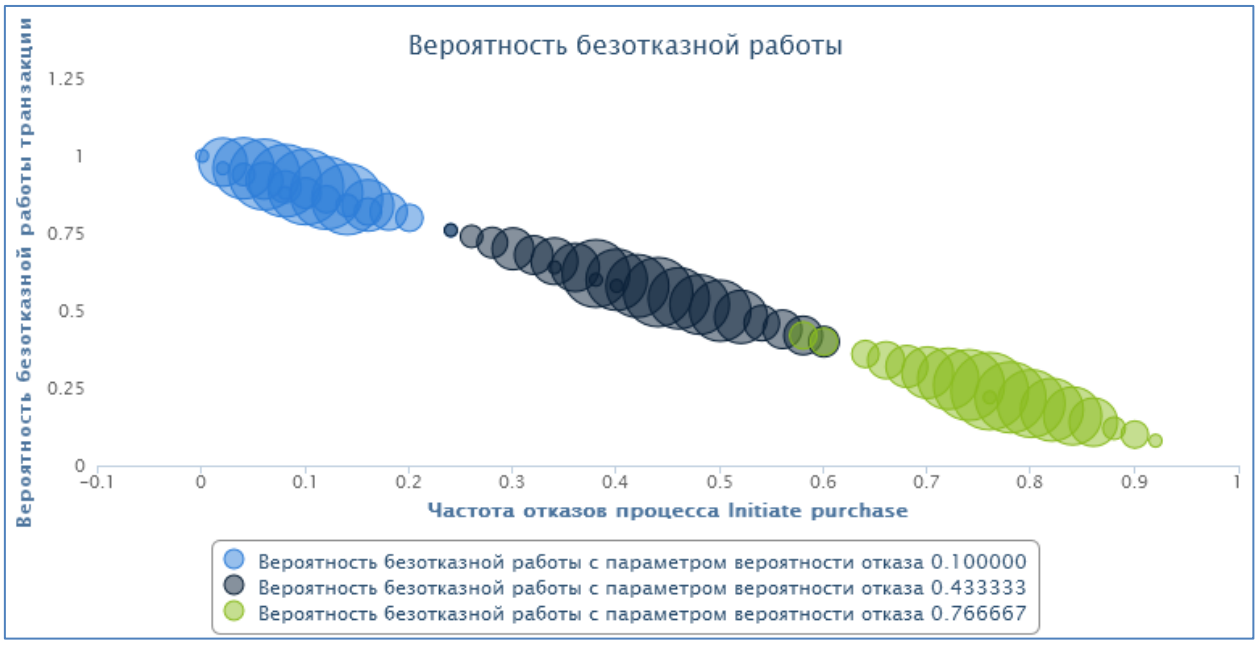


Рисунок 21 – Корреляционное поле по результатам исследования чувствительности

Единственным отличающимся процессом был выявлен процесс «Send Payment notif.». Как можно отметить по рис. 22, какой бы ни была частота его отказов, вероятность отказа бизнес-транзакции не изменяется. Таким образом, процесс «Send Payment notif» не влияет на целевой показатель надежности и может быть исключен из рассмотрения.

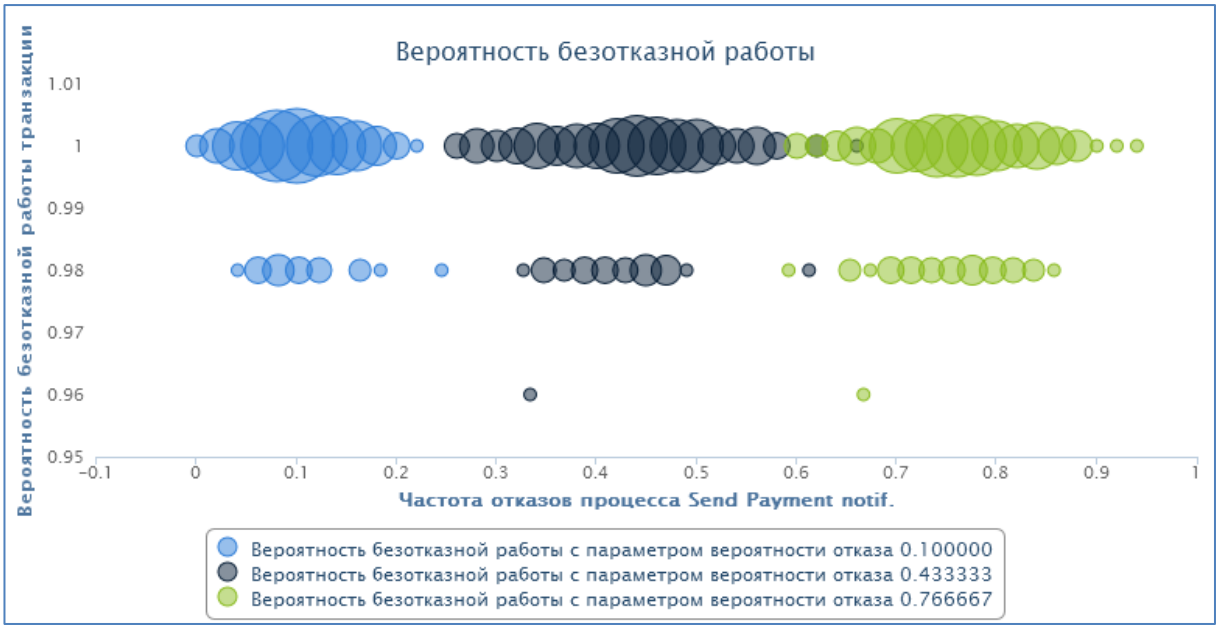
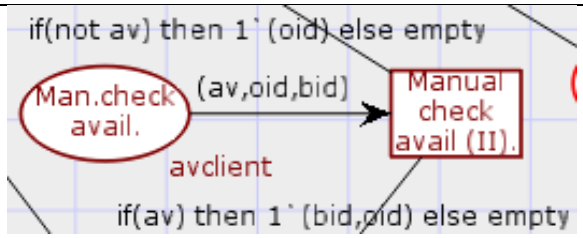


Рисунок 22 – Влияния процесса «Send Payment notif.» на стабильность бизнес-транзакции

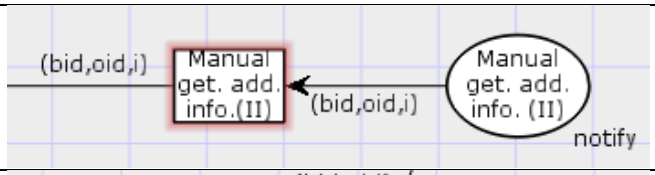
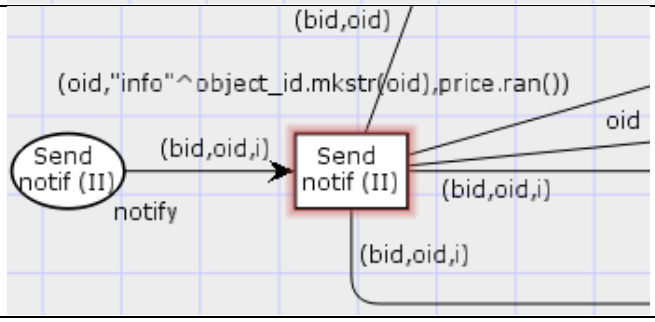
Так как все исследуемые процессы, кроме «Send Payment notif», одинаково влияют на стабильность работы бизнес-транзакции (угол наклона линии тренда графиков одинаков и это отношение может быть выражено через уравнение $y = 1 - x$, где y – вероятность безотказной работы, а x – вероятность отказа элемента), то имеет смысл повышать стабильность работы самых ненадежных элементов.

Введем резервирование элементов «Manual check avail.», «Send Info», «Manual get. add. Info.» и «Send notif» через добавление восстанавливающих сетей (см. табл. 17), что позволит снизить вероятность отказов при выполнении этих операций.

Таблица 17 – Восстанавливающие сети для ненадежных процессов

Операция	**Схема восстанавливающей композиции**	**Интерфейс входящей операции**
Manual check avail.		(av,bid,oid)

Продолжение таблицы 17

Операция	Схема восстанавливающей композиции	Интерфейс входящей операции
Manual get. add. Info.		(bid,oid,i)
Send notif		(bid,oid,i)

Вероятность отказа восстанавливающих сетей положим равной вероятности отказа восстанавливаемых операций. По результатам четвертого раунда эксперимента вероятность безотказной работы вероятность отказа достигла показателя 0,7749[1], что на 75% выше показателя без резервирования основных сбойных элементов.

Таблица 18 – Результаты четвертого раунда эксперимента

Показатель	Значение
Атомарность (в операциях)	0,9912
Атомарность (в общем)	0,9995
Вероятность избежания аварий	0,9997
Разрешимость отказов	0,9558
Вероятность безотказной работы	0,7749
Вероятность безаварийной работы	0,9995
Частота отказов	0,2251
Частота аварий	0,0005
Наработка на отказ	3,7784
Наработка на аварию	49,0000
Интенсивность отказов	0,2974
Интенсивность аварий	0,0204

Однако полученные результаты в виде показателя вероятности безотказной работы все еще не удовлетворяют поставленным требованиям надежности, и меры по повышению надежности отдельных элементов следует продолжать. Рассмотрим текущие показатели работы процессов, полученные по результатам эксперимента (см. табл. 19).

[1] В настройках ПКБТ в целях более точного анализа точность вывода результатов была повышена с двух до четырех десятых.

Таблица 19 – Вероятность отказа процессов

Показатель	Значение
Initiate purchase	0,0481
Check avai.	0,0203
Get Info	0,0089
Notice operator	0,0365
Manual check avail. [1]	0,0825
Send notif	0,0215
Reserve object	0,0055
Get add. information	0,0062
Send Info	0,0251
Manual get. add. info.	0,0274
Check avai2	0,02

Информационных ресурсы компании (витрина, магазин, внутренняя система), как уже говорилось, функционируют на сервере провайдера, настройки и стабильность работы которого существенно влияют на стабильность работы отдельных операций. Теоретически, возможно дублирование всех серверных ресурсов, поэтому для таких процессов, вероятность отказа которых превышает 1% определим восстанавливающие последовательности по технологии, представленной в таблице 17, а для тех процессов, которые уже были зарезервированы, зарезервируем критические процессы в их восстанавливающих последовательностях и проведем пятую итерацию эксперимента. Сервисы, которые работают на стороне оператора – «Notice operator» и «Manual get. add. info.» подвергать резервированию на данном шаге не будем, так как это приводит к удорожанию процедуры повышения надежности.

По результатам этого прогона вероятность безотказной работы бизнес-транзакции (см. табл. 20) достигла значения 0,9143, а атомарность – 0,9991, что отвечает поставленным целям.

Таблица 20 – Результаты пятого раунда эксперимента

Показатель	Значение
Атомарность (в операциях)	0,991
Атомарность (в общем)	0,9991
Вероятность избежания аварий	0,9987
Разрешимость отказов	0,8416
Вероятность безотказной работы	0,9143
Вероятность безаварийной работы	0,9991
Частота отказов	0,0857

[1] Для уже зарезервированных процессов показана вероятность отказа с учетом вероятности отказа их восстанавливающих сетей

Продолжение таблицы 20

Показатель	Значение
Частота аварий	0,0009
Наработка на отказ	14,4396
Наработка на аварию	46,9167
Интенсивность отказов	0,0975
Интенсивность аварий	0,0222

На каждые 50 запусков происходит максимум 13 отказов (см. рис. 23), а в среднем 3.1. Диаграмма частот отказов уже сдвигается от вида биномиального распределения в сторону распределения Пуассона, это означает, что события отказов начинают происходить сравнительно редко.

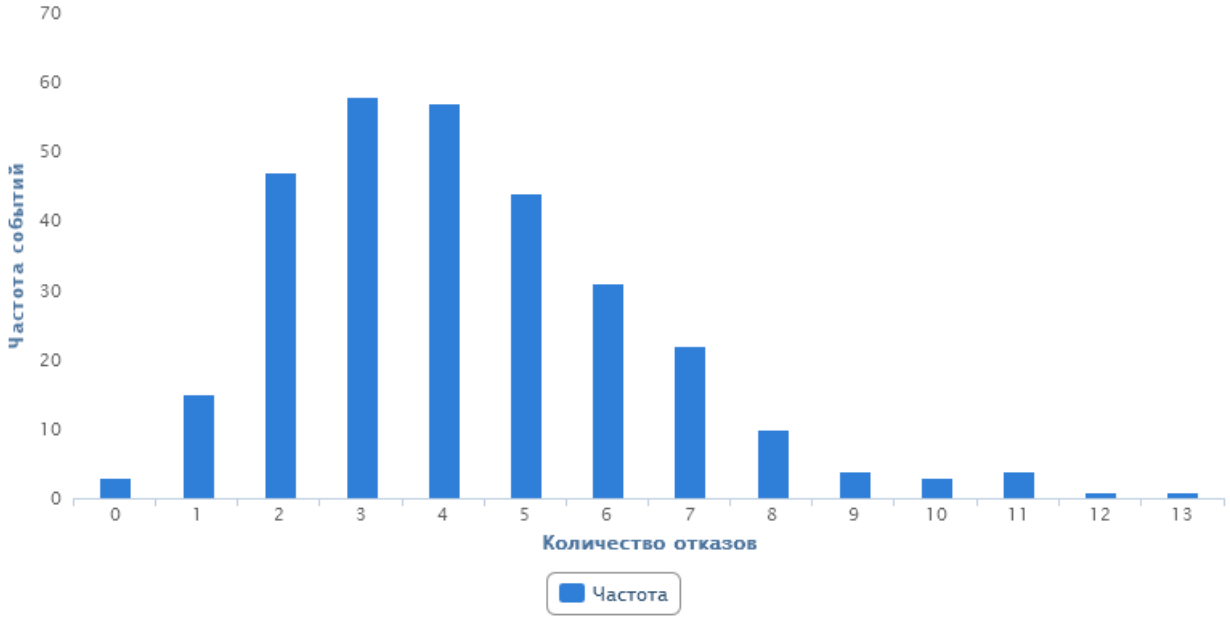


Рисунок 23 – Результаты пятого раунда эксперимента (количество отказов на 50 запусков)

Для получения окончательной информации о надежности работы бизнес-транзакции в условиях внешнего воздействия, был проведен эксперимент с учетом влияния желаний клиентов, отказов внешних сервисов и вероятной недоступности объектов. Как видно из таблицы 21 вероятность безотказной работы бизнес-транзакции составляет 0,4531. То есть только в 45% случаев клиент, инициировавший бизнес-транзакцию, сможет получить товар. В остальных случаях товар может оказаться недоступным, клиент откажется от его оплаты или произойдет маловероятное событие – отказ платежной или sms-системы.

Таблица 21 – Результаты шестого раунда эксперимента

Показатель	Значение
Атомарность (в операциях)	0,9933
Атомарность (в общем)	0,9997
Вероятность избежания аварий	0,9993
Разрешимость отказов	0,8883
Вероятность безотказной работы	0,4531
Вероятность безаварийной работы	0,9997
Частота отказов	0,5469
Частота аварий	0,0003
Наработка на отказ	0,8608
Наработка на аварию	49
Интенсивность отказов	1,2665
Интенсивность аварий	0,0204

Таким образом, проведенный анализ позволяет определить, что для достижения следующих показателей исследуемой бизнес-транзакции:

Показатель	Значение
Вероятность безотказной работы (внутренняя надежность)	0,9143
Вероятность безотказной работы (общая надежность)	0,4531

необходимо:

- Увеличить количество операторов, обслуживающих бизнес-транзакцию в 3 раза.
- Провести дублирование всех сервисов компании, находящихся на сервере провайдера.

Проведем анализ устойчивости полученной бизнес-транзакции с помощью алгоритма на цепях Маркова, предложенного в 2.7.

Для этого необходимо инициировать 4-й тип анализа, а в настройках эксперимента указать те позиции, которые необходимо учитывать при анализе маркировок. Исключим из схемы бизнес-транзакции возможность появления недетерминированных данных в позициях хранения (т.е. исключим случайную генерацию данных) и потребуем однократного выполнения в процессе одного запуска имитации. Это позволит минимизировать количество возможных маркировок без ущерба для результатов анализа (подробнее об специальных требованиях анализа устойчивости см. в Приложении А).

Результат анализа устойчивости показал, что бизнес-транзакция успешно выполняется (см. рис. 24).

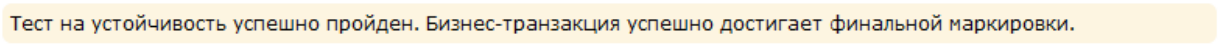


Рисунок 24 – Результат анализа устойчивости бизнес-транзакции

Построенный граф по матрице переходных вероятностей позволяет убедится самостоятельно (см. рис. 25).

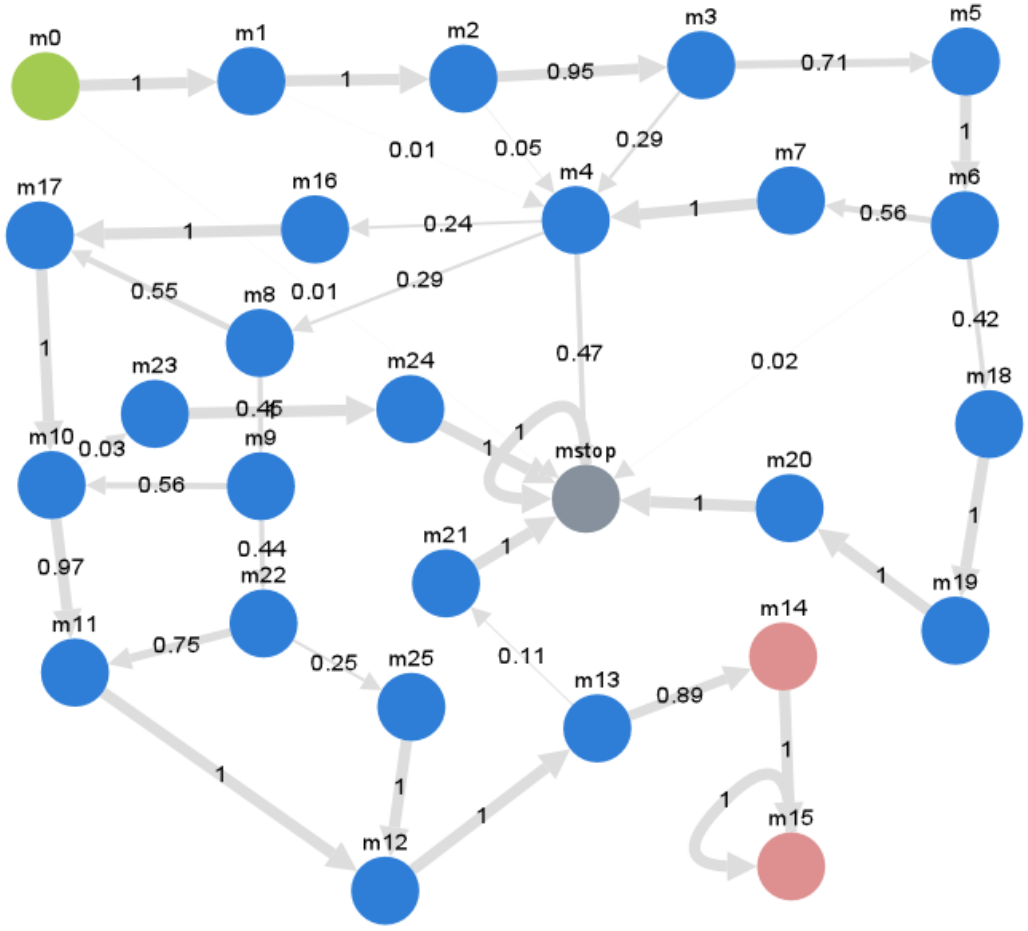


Рисунок 25 – Граф переходов состояний (маркировок) бизнес-транзакций

Выводы по главе 3

Для демонстрации эффективности работы программного комплекса было проведено сравнение некоторых распространенных аналитических методов расчета надежности дискретных систем и программного комплекса.

Рассмотрено решение практической задачи проектирования бизнес-транзакции с заданной надежностью с помощью программного комплекса и предложенной методики. Бизнес-транзакция функционирует в среде сложного интернет-магазина, выполняется 8 независимыми участниками и включает 18 операций, работа который влияет на надежность всей системы. Проанализирован характер связанности участников и операций, что позволило предложить способы построения компенсирующих и восстанавливающих сетей. Дано графическое описание бизнес-транзакции с помощью окрашенных сетей Петри.

Показан процесс последовательного применения программного комплекса для выявления «слабых» мест надежности вынесения рекомендаций по их устранению и повышению надежности системы в целом. Это позволяет сделать вывод о целесообразности использования ПКБТ для анализа надежности проектируемых бизнес-транзакций.

Заключение

Несмотря на то, что термин «бизнес-транзакция» употребляется сравнительно редко, в настоящее время сложно не найти черты этого явления во всех современных экономических информационных системах. В какой бы прикладной области они ни работали и какого бы масштаба ни были, они сталкиваются с необходимостью управлять неделимыми последовательностями операций, которые могут выполняться различными участниками. Будь то аппаратные, программные системы, обслуживающий персонал или клиенты компании. А повышение уровня открытости систем, облегчение их интеграции и стремление к электронному взаимодействию вынуждает предприятия существенно усложнять используемые системы и выполняемые бизнес-процессы, привлекая в них не только внутренних исполнителей, но и сторонние организации.

В данном исследовании предложена методика анализа надежности бизнес-транзакций, которая основывается на графическом представлении и последующем имитационном моделировании. Предложенные количественные показатели надежности позволяют всесторонне оценивать надежность в понимании того, что бизнес-транзакция – это и взаимодействующие программы, бизнес-процесс, выполняемый на предприятии, и неразрывное согласованное движение всей системы от начального состояния к финальному.

Предложен набор примитивов на базе окрашенных сетей Петри, который описывает уникальные особенности и элементы сервис-ориентированной архитектуры. Примитивы в совокупности составляют специальный инструмент моделирования, который позволяет аналитически представлять и проектировать подобные системы. С помощью этого инструмента было дано формальное определение бизнес-транзакции, а также показана возможность определения ее надежности, исследуя только структуру.

В основе разработанной методики лежит системный подход к пониманию сущности бизнес-транзакции. Так как начальные этапы методики совпадают с трудноформализуемыми этапами системного анализа [146], то разработанное программное обеспечение только частично автоматизирует ее применение в самой трудоемкой части: проведении имитационного моделирования, расчете показателей, проведении статистического анализа. При этом программный комплекс ориентирован на применение аналитиками, специалистами по проектированию бизнес-процессов и нетехническим персоналом, так как интерфейс про-

граммы сравнительно прост, а многие технические и математические проблемы скрыты от пользователя.

Отметим следующие результаты работы:

- Формализовано понятие бизнес-транзакции: предложено рассматривать единичный акт (неделимую последовательность операций) работы процессно-ориентированной распределенной системы как самостоятельный объект, отражающий цели существования этой системы и внутренние характеристики ее функционирования.
- Установлено, что бизнес-транзакция характеризуется уникальными свойствами, разделяется на несколько типов, имеет существенные отличия от системных транзакций и может определять концепцию построения распределенной системы, ее выполняющей.
- Предложено рассматривать бизнес-транзакцию как сложную разнородную систему, имеющую тройственную природу, что позволяет интегрировать подходы к оценке надежности из разных прикладных областей. Выделены свойства бизнес-транзакции как сложной системы, обоснована целесообразность и разработана методика анализа ее надежности с помощью имитационного моделирования.
- Разработана система показателей надежности бизнес-транзакции. Многие стандартные показатели надежности технических или программных средств для бизнес-транзакции могут быть неприменимы. Поэтому на основании ее системных свойств и тройственной природы строится подход к оценке ее надежности в виде комплекса параметров, функционирование в заранее заданных границах которых и позволит сказать разработчику, что конкретная исследуемая им бизнес-транзакция – надежна.
- Разработан инструмент, который позволяет моделировать распределенные сервис-ориентированные системы, механизмы композиции сервисов, их отказов, операций восстановления и компенсации и другие высокоуровневые структуры, описывающие элементы сервис-ориентированной среды, а также некоторые важные идеи функционирования и имитационного моделирования бизнес-транзакции.
- Предложена модель бизнес-транзакции на базе разработанного инструмента моделирования. Вместе с привлеченной теорией цепей Маркова удалось выразить на языке окрашенных сетей Петри способы расчета предложенных показателей надежности.

- Предложен алгоритм обнаружения тупиковых ситуаций и бесконечных циклов на основе представления модели бизнес-транзакции в виде цепи Маркова.
- Предложена методика анализа надежности бизнес-транзакции с учетом результатов исследования ее системных свойств, использования разработанного аппарата моделирования и предложенных показателей, описывающих ее тройственную природу.
- Разработано программное обеспечение для анализа надежности бизнес-транзакции с использованием разработанного аппарата моделирования и методики надежности.
- Успешно решены следующие задачи:
    - выявлены проблемы надежности проектируемой бизнес-транзакции обеспечения продаж, выполняющейся в среде нескольких предприятий и инициируемой компанией «ООО Группа компаний Инвест38». Предложены пути повышения надежности этой бизнес-транзакции. Программный комплекс внедрен на предприятии и успешно используется для разработки новых распределенных систем.
    - выявлены проблемы надежности внедряемого модуля «Оценка стоимости земельных участков» в автоматизированную информационную систему компании «ООО Группа компаний Инвест38». Рекомендации, полученные с помощью разработанного программного комплекса, позволили внести изменения в модуль на этапе его проектирования и тем самым снизить расходы на его сопровождение и устранение проблем надежности.

Таким образом, можно сделать вывод, что поставленные задачи исследования в целом выполнены.

Используемые сокращения

АИС	—	автоматизированная информационная система
БД	—	база данных
БТ	—	бизнес-транзакция
ПКБТ	—	программный комплекс моделирования бизнес-транзакций
ПО	—	программное обеспечение
ПОИС	—	процессно-ориентированная информационная система
СОА	—	сервис-ориентированная архитектура
СУБД	—	система управления базами данных
ACID	—	Atomicity, Consistency, Isolation, Durability – атомарность, непротиворечивость, изолированность, долговечность – набор свойств системных транзакций
ARIS	—	Architecture of Integrated Information Systems – методология моделирования бизнес-процессов организаций
B2B	—	Business-to-Business – взаимодействие юридических лиц друг с другом
B2C	—	Business-to-Customer – взаимодействие предприятия и клиента
BP	—	Business Process – бизнес-процесс
BPEL	—	Business Process Execution Language – язык формального описания и исполнения бизнес-процессов
BPM	—	Business Process Management – методология управления бизнес-процессами
BPMN	—	Business Process Model and Notation – нотация моделирования бизнес-процессов
BPMS	—	Business Process Management System – система управления бизнес-процессами, реализующая методологию BPM
CP nets или CPN	—	Coloured Petri nets – окрашенные сети Петри
DFD	—	Data Flow Diagrams – диаграммы потоки данных
EAI	—	Enterprise Application Integration – интеграция приложений на предприятии
EPC	—	Event-driven Process Chain – блок-схемы для бизнес-моделирования
HTTP	—	HyperText Transfer Protocol – протокол прикладного уровня передачи данных
IDEF	—	Icam DEFinition – методология моделирования сложных систем
PN nets	—	Position/Transition – сети Петри типа "позиция-переход"
QoS	—	Quality of Service – качество обслуживания
SADT	—	Structured Analysis and Design Technique – методология структурного анализа и проектирования

SLA	—	Service Level Agreement – соглашение об уровне обслуживания
SOA	—	Service Oriented Architecture – сервис-ориентированная архитектура
SOAP	—	Simple Object Access Protocol – XML-ориентированный протокол обмена структурированными сообщениями в распределенной среде
UDDI	—	Universal Description, Discovery and Integration – протокол для поддержки работы реестра описаний сетевых служб
UML	—	Unified Modeling Language – унифицированный язык моделирования
URI	—	Unified Resource Identifier – универсальный идентификатор ресурса
WF	—	WorkFlow – поток работ
WFMS	—	Workflow Management System – система управления потоком работ
WS	—	Web Service – сетевая служба, веб-служба, веб-сервис
WSDL	—	Web-service description language – стандарт описания интерфейса сетевой службы
XML	—	eXtensible Markup Language – расширяемый язык разметки документов

Список литературы

1. Репин, В. В. Бизнес-процессы компании: построение, анализ, регламентация / В. В. Репин. – М. : РИА «Стандарты и качество», 2007. – 240 с.

2. Репин, В. В. Процессный подход к управлению. Моделирование бизнес-процессов / В. В. Репин, В. Г. Елиферов. – М. : РИА «Стандарты и качество», 2005. – 405 с.

3. Самуйлов, К. Е. Основы формальных методов описания бизнес-процессов / К. Е. Самуйлов, Н. В. Серебренникова, А. В. Чукарин, Н. В. Яркина. – М. : РУДНА, 2008. – 130 с.

4. van der Aalst, W. Process-Aware Information Systems: Lessons to Be Learned from Process Mining / Wil M.P. van der Aalst // Transactions on Petri Nets and Other Models of Concurrency II / Kurt Jensen, Wil M.P. van der Aalst. – Springer, 2009. – С. 1-26.

5. ter Hosftede, A. H. Process-Aware Information Systems: bridging people and software through process technology / Arthur H.M. ter Hosftede, Marlon Dumas, Wil M.P. van der Aalst. – John Wiley & Sons, Inc, 2005. – 409 с.

6. Хоп, Г. Шаблоны интеграции корпоративных приложения / Г. Хоп, Б. Вульф. – М. : ООО «И.Д. Вильямс», 2007. – 672 с.

7. Juric, M. B. SOA Approach to Integration / Matiaz B. Juric. – Birmingham : Pact Publishing Ltd., 2007. – 366 с.

8. Linthicum, D. S. Enterprise Application Integration / David S. Linthicum. – Addison Wesley, 1999. – 400 с.

9. La Rosa, M. Managing Variability in Process-Aware Information Systems / M. La Rosa. – Brisbane : Faculty of Science and Technology Queensland University of Technolog, 2009. – 165 с.

10. Manouvrier, B. Application Integration: EAI, B2B, BPM and SOA / Bernard Manouvrier, Laurent Ménard. – John Wiley & Sons, 2008. – 224 с.

11. Linthicum, D. S. Next Generation Application Integration: From Simple Information to Web Services / David S. Linthicum. – Addison Wesley, 2003. – 512 с.

12. Kaye, D. Loosely Coupled: The Missing Pieces of Web Services / Doug Kaye. – RDS Press , 2003. – 334 с.

13. Доррер, М. Г. Оценка параметров моделей бизнес-систем с использованием программного обеспечения класса workflow / М. Г. Доррер, А. А. Некрасова // Материалы конференции «Проблемы информатизации региона». – Красноярск, 2003. – С. 122-126.

14. Roshen, W. SOA-Based Enterprise Integration: a step-by-step guide to service-based aplication integration / Waseem Roshen. – The MCGraw Hill, 2009. – 364 с.

15. Артамонов, И. В. Разработка распределенных сервисно-ориентированных программных средств / И. В. Артамонов. – Иркутск : БГУЭП, 2012. – 130 с.

16. Ruh, W. A. Enterprise Application Integration / William A. Ruh, Francis X. Maginnis, William J. Brown. – Wiley, 2001. – 211 с.

17. Фаулер, М. Архитектура корпоративных программных приложений / М. Фаулер. – М. : Издательский дом «Вильямс», 2006. – 544 с.

18. Артамонов, И. В. Слабое связывание как фактор эволюции технологий интерграции распределенных систем / И. В. Артамонов // Материалы конференции «Инновации на основе информационных и коммуникационных технологий». – Москва, 2014. – С. 294-297.

19. Pulier, E. Understanding Enterprise SOA / Eric Pulier, Hugh Taylor. – Greenwich : Manning, 2006. – 242 с.

20. Артамонов, И. В. Бизнес-транзакции: характеристики и отличительные особенности / И. В. Артамонов // Бизнес-информатика. – 2012. – № 2(20). – С. 29-34.

21. Артамонов, И. В. Компонентно- и сервисно-ориентированная модель распределенных программных средств / И. В. Артамонов // Применение математических и информационных технологий в экономике. – 2009. – № 8. – С. 55-66.

22. Шорт, С. Разработка XML Web-сервисов средствами Microsoft.NET / Скотт Шорт. – СПб : БХВ-Петербург, 2003. – 480 с.

23. Дубова, Н. SOA: подходы к реализации [Электронный ресурс] / Наталья Дубова // Открытые системы. – 2004. – Режим доступа: http://www.osp.ru/os/2004/06/184450/ (дата обращения: 15.10.2008).

24. Booth, D. Web Services Architecture [Электронный ресурс] / D. Booth, H. Haas, F. McCabe, E. Newcomer, M. Champion, C. Ferris, D. Orchard // W3C. – W3C Working Group, 2004. – Режим доступа: http://www.w3.org/TR/ws-arch/ (дата обращения: 12.05.2013).

25. Cervantes, H. Service-oriented software system engeenering: challenges and practices / Humberto Cervantes, Richard S. Hall // Материалы конференции «Techinal Concepts of Service Orientation». – London, 2005. – С. 1-26.

26. Erl, T. Service-Oriented Architecture: Concepts, Technology, and Design / Thomas Erl. – Prentice Hall PTR, 2005. – 792 с.

27. Cohen, F. FastSOA / Frank Cohen. – Elsevier Inc, 2007. – 278 с.

28. Bieberstein, N. Executing SOA: A Practical Guide for the Service-Oriented Architect / Norbert Bieberstein, Robert G. Laird, Keith Jones, Tilak Mitra. – IBM Press, 2008. – 240 с.

29. Booch, G. Software Architecture and the UML. Rational Software Report [Электронный ресурс] / Grady Booch. – Режим доступа: http://www.uml.org.cn/UMLApplication/pdf/booch.pdf (дата обращения: 10.09.2012).

30. MacKenzie, C. M. Reference Model for Service Oriented Architecture 1.0 [Электронный ресурс] / C. Matthew MacKenzie, Ken Laskey, Francis McCabe, Peter F Brown, Rebekah Metz. – OASIS, 2006. – Режим доступа: https://www.oasis-open.org/committees/download.php/19679/soa-rm-cs.pdf (дата обращения: 15.10.2012).

31. Carter, S. The New Language of Business: SOA & Web 2.0 / Sandy Carter. – IBM Press , 2007. – 299 с.

32. Gao, T. Y. The Complete Reference To Professional SOA With Visual Studio 2005 / Tom Yuan Gao. – LULU Press, 2007. – 788 с.

33. Rosen, M. Applied SOA: Service-Oriented Architecture and Design Strategies / Mike Rosen, Boris Lublinsky, Kevin T. Smith, J. Marc Balcer. – Wiley Publishing, Inc., 2008. – 662 с.

34. Brown, P. C. Implementing SOA: Total Architecture in Practice / Paul C. Brown. – Addison Wesley Professional, 2008. – 736 с.

35. McGovern, J. A Practical Guide to Enterprise Architecture / James McGovern, Scott Ambler, Michael Stevens, James Linn, Vikas Sharan, Elias Jo. – Prentice Hall PTR, 2003. – 336 с.

36. Steen, M. W. Service-oriented software system engeenering: challenges and practices / Maarten W.A. Steen, Patrick Strating, Marc M. Patrick, Hugo W.L. ter Doest, Maria-Eugenia Iacob // Материалы конференции «Service-oriented enterprise architecture». – London, 2005. – С. 132-154.

37. Chou, D. SOA with .NET and Windows Azure / David Chou, John deVadoss, Thomas Erl, Nitin Gandhi, Hanu Kommalapati, Brian Loesgen, Christoph Schittko, Herbjörn Wilhelmsen, Mickey Williams. – Prentice Hall, 2010. – 870 с.

38. Benatallah, B. Service-oriented software system engeenering: challenges and practices / Boualem Benatallah, Remco M. Dijkman, Marlon Dumas, Zakaria Maamar // Материалы конференции «Service Composition: Concepts, Techniques,Tools and Trends». – London, 2005. – С. 48-67.

39. Hewitt, E. Java SOA Cookbook / Eben Hewitt. – O'Reilly, 2009. – 715 с.

40. Holley, K. 100 SOA Questions Asked and Answered / Kerrie Holley, Ali Arsanjani. – Prentice Hall, 2010. – 242 с.

41. Jocuttis, N. M. SOA in Practice / Nicloai M. Jocuttis. – Sebastopol : O'Reilly, 2007. – 324 с.

42. Беленкович, В. Логическая структура понятия сервисов в рамках SOA / В. Беленкович, Т. Горшков // Сетевой журнал. – 2005. – № 4.

43. Catts, A. Business Process Managment enabled by SOA / Anthony Catts, Joseph St.Clair. – IBM Press, 2009. – 82 с.

44. Brocke, J. Handbooks on business process managment: Introduction, Methods and Informations Systems / J.Vom Brocke, M. Rosemann. – Springer, 2010. – 612 с.

45. Smith, H. Business Process Management: The Third Wave / Howard Smith, Peter Fingar. – Meghan-Kiffer Press, 2003. – 311 с.

46. van der Aalst, W. In Proceedings of the Business Process Management: International Conference / Wil M.P. van der Aalst, Arthur H.M. ter Hofstede, Mathias Weske // Материалы конференции «Business process management: A survey». – 2003. – С. 1-12.

47. Weske, M. Business Process Managment: Concepts, Languages, Architectures / Mathias Weske. – Springer, 2007. – 368 с.

48. H. Business Process Management: The Third Wave / Howard Smith, Peter Fingar. – Meghan-Kiffer Press, 2003. – 311 с.

49. K.L.Ko, R. A computer scientist introductory guide to business process managment / Ryan K.L.Ko // Crossroads. – 2009. – Т. 15, № 4.

50. Биберштейн, Н. Компас в мире сервис-ориентированной архитектуры / Норберт Биберштейн, Сенджей Боуз, Марк Фиаммант, Кейт Джонс, Роун Ша. – М. : Кудиц-Пресс, 2007. – 256 с.

51. Вендров, А. М. Проектирование программного обеспечения экономических информационных систем / А. М. Вендров. – М. : Финансы и статистика, 2006. – 544 с.

52. Грекул, В. И. Проектирование информационных систем / В. И. Грекул, Г. Н. Денищенко, Н. Л. Коровкина. – М. : Интуит, 2005. – 304 с.

53. Мишенин, А. И. Теория экономических информационных систем / А. И. Мишенин. – М. : Финансы и статистика, 2002. – 240 с.

54. Артамонов, И. В. Описание бизнес-процессов: вопросы стандартизации. / И. В. Артамонов // Прикладная информатика. – 2011. – № 3. – С. 20-28.

55. Bell, M. Service-Oriented Modeling Service Analysis Design and Architecture / Michael Bell. – John Wiley & Sons Ltd, 2008. – 366 с.

56. Уткин, В. Б. Информационные системы в экономике / В. Б. Уткин, К.В. Балдин. – М. : Издательский центр «Академия», 2004. – 288 с.

57. Zafar, B. Conceptual Modelling of Adaptive Web Services based on High-level Petri Nets: PhD Thesis / Bassam Zafar. – De Montfort University, 2008. – 188 с.

58. Latchem, S. Service-Oriented Design Process Using UML / Steve Latchem, David Piper // Service-oriented software system engineering : challenges and practices / Zoran Stojanovic, Ajantha Dahanayake. – Idea Group Publishing, 2005. – С. 88-108.

59. Семакин, И. Г. Информационные системы и модели / И. Г. Семакин, Е. К. Хеннер. – М. : Бином, 2005. – 303 с.

60. Артамонов, И. В. Современные стандарты описания и исполнения бизнес-процессов / И. В. Артамонов // Современные стандарты описания и исполнения бизнес-процессов. – Иркутск : БГУЭП, 2010. – № 9. – С. 5-31.

61. Booch, G. Unified Modeling Language User Guide / Grady Booch, James Rumbaugh, Ivar Jacobson. – Addison-Wesley, 2005. – 496 с.

62. Самуйлов, К. Е. Бизнес-процессы и информационные технологии в управлении телекоммуникационными компаниями / К. Е. Самуйлов, А. В. Чукарин, Н. В. Яркина. – Москва : Альпина-Паблишерз, 2009. – 442 с.

63. Lohmann, N. Petri Net Transformations for Business Processes – A Survey / Niels Lohmann, Eric Verbeek, Remco Dijkman // Transactions on Petri Nets and Other Models of Concurrency II / Kurt Jensen, Wil M.P. van der Aalst. – Springer, 2009. – С. 46-63.

64. Graham, I. Requirements Modelling and Specification for Service Oriented Architecture / Ian Graham. – John Wiley & Sons Ltd, 2008. – 301 с.

65. Hruz, B. Modeling and Control of Discrete-event Dynamic Systems / B. Hruz, M. C. Zhou. – Springer, 2007. – 341 с.

66. van der Aalst, W. Business process management demystified: A tutorial on models, systems and standards for workflow managment / Wil M.P. van der Aalst // Lectures on Concurrency and Petri Nets. – 2004. – № 3098. – С. 1-65.

67. Campos, J. On the Integration of UML and Petri Nets in Software Development / J. Campos, J. Merseguer // Petri Nets and Other Models of Concurrency / Susanna Donatelli, P.S. Thiagarajan. – Springer, 2006. – С. 19-36.

68. Denaro, G. Petri Nets and Software Engineering / Giovanni Denaro, Mauro Pezze // Lectures on Concurrency and Petri Nets: Advances in Petri Nets / Jörg Desel, Wolfgang Reisig, Grzegorz Rozenberg. – Springer, 2005. – С. 439-467.

69. Zurawski, R. Petri Nets and Industrial Applications: A Tutorial / Richard Zurawski, MengChu Zhou // IEEE. Transactions on industrial electronic. – 1994. – Т. 41. – С. 567-583.

70. Jensen, K. An Introduction to the Theoretical Aspects of Coloured Petri Nets / Kurt Jensen // A Decade of Concurrency, Lecture Notes in Computer Science / J.W. de Bakker, W.-P. de Roever, G. Rozenberg. – Springer-Verlag, 1994. – С. 230-272.

71. Smith, E. Principles of High-Level net Theory / E. Smith // Lectures on Petri Nets I: Basic Models / Wolfgang Reisig, Grzegorz Rozenberg. – Springer, 1998. – С. 174-210.

72. Silva, M. Petri nets and Production Systems / M. Silva, E. Teruel, R. Valette, H. Pingaud // Lectures on Petri Nets II: Applications / Wolfgang Reisig, Grzegorz Rozenberg. – Springer, 1998. – С. 85-124.

73. van der Aalst, W. The Application of Petri Nets to Workflow Management / W.M.P. van der Aalst // Journal of Circuits, Systems and Computers. – 1998. – Т. 8, № 1. – С. 21-66.

74. van der Aalst, W. Three good reasons for using a petri-net-based workflow management system / W. M. P. van der Aalst // Материалы конференции «In Proceedings of the International Working Conference on Information and Process Integration in Enterprises». – 1996. – С. 179-201.

75. Massuthe, P. Operating Guidelines for Services: Dissertation / Peter Massuthe. – University Press Facilities, 2009. – 266 с.

76. Wolf, K. Does My Service Have Partners? / Karsten Wolf // Transactions on Petri Nets and Other Models of Concurrency II / Kurt Jensen, Wil M.P. van der Aalst. – Springer, 2009. – С. 152-171.

77. Popescu, C. A Petri net-based approach to incremental modelling of flow and resources in service-oriented manufacturing systems / Corina Popescu, M. Cavia Soto, Jose L. Martinez Lastraa // International Journal of Production Research. – 2012. – Т. 50, № 2. – С. 325-343.

78. Bernardi, S. Petri Nets and Dependability / Simona Bernardi, Andrea Bobbio, Susanna Donatelli // Lectures on Concurrency and Petri Nets / Jörg Desel, Wolfgang Reisig, Grzegorz Rozenberg. – Springer, 2005. – С. 125-179.

79. Kohler, M. Web Service Orchestration with Super-Dual Object Nets / Michael Kohler, Heiko Rolke // Petri Nets and Other Models of Concurrency / Jetty Kleijn, Alex Yakovlev. – Springer, 2007. – С. 263-280.

80. Men, P. Utilizing Fuzzy Petri Net for Choreography Based Semantic Web Services Discovery / P. Men, Z. Duan, B. Yu // Petri Nets and Other Models of Concurrency / Jetty Kleijn, Alex Yakovlev. – Springer, 2007. – С. 362-380.

81. Hartonas, C. South-East European Network on Formal Methods / Chrysafis Hartonas // Материалы конференции «A Combined Open Petri Net and Process Algebraic approach to Message-Passing Services». – 2007.

82. Jensen, K. Coloured Petri Nets modeling and validation of concurrent systems / Kurt Jensen. – Springer, 2009. – 384 с.

83. Bruni, R. Extending the Zero-Safe Approach to Coloured, Reconfigurable and Dynamic Nets / R. Bruni, H. Melgratti, U. Montanari // Lectures on Concurrency and Petri Nets / Jörg Desel, Wolfgang Reisig, Grzegorz Rozenberg. – Springer, 2005. – С. 291-327.

84. Recalde, L. Petri Nets and Manufacturing Systems: An Examples-Driven Tour / L. Recalde, M. Silva, J. Ezpeleta, E. Teruel // Lectures on Concurrency and Petri Nets / Jörg Desel, Wolfgang Reisig, Grzegorz Rozenberg. – Springer, 2005. – С. 742-789.

85. Acu, B. Compensation in Workflow Nets / Baver Acu, Wolfgang Reisig // Petri Nets and Other Models of Concurrency / Susanna Donatelli, P.S. Thiagarajan. – Springer, 2006. – С. 65-83.

86. Kindler, E. Model-Based Software Engineering and Process-Aware Information Systems / E. Kindler // Transactions on Petri Nets and Other Models of Concurrency II / Kurt Jensen, Wil M.P. van der Aalst. – Springer, 2009. – С. 27-46.

87. Reichert, M. Flexibility in Process-Aware Information Systems / M. Reichert, S. Rinderle-Ma, P. Dadam // Transactions on Petri Nets and Other Models of Concurrency II / Wil M.P. van der Aalst, Kurt Jensen. – Springer, 2009. – С. 115-135.

88. van de Aalst, W. Soundness of Workflow Nets with Reset Arcs / Wil M.P. van de Aalst // Transactions on Petri Nets and Other Models of Concurrency III / Kurt Jensen, Jonathan Billington, Maciej Koutny. – Springer, 2009. – С. 50-70.

89. ГОСТ Р ИСО/МЭК9126-93. Оценка программного продукта. Характеристики качества и руководства по их применению [Электронный ресурс]. – ГОССТАНДАРТ РОССИИ, 1994. – Режим доступа: http://www.gametest.ru/doc/sw/9126_93.pdf (дата обращения: 05.04.2013).

90. Липаев, В. В. Программная инженерия. Методологически еосновы / В. В. Липаев. – М. : ТЕИС, 2006. – 608 с.

91. Острейковский, В. А. Теория Надежности / В. А. Острейковский. – М. : Высш.шк., 2003. – 463 с.

92. Голинкевич, Т. А. Прикладная теория надежности / Т. А. Голинкевич. – М. : Высшая школа, 1977. – 160 с.

93. Авдуевский, В. С. Надежность и эффективность в технике: Справочник. В 10 т. Т.1: Методология. Организация. Терминология / В. С. Авдуевский. – М. : Машиностроение, 1986. – 224 с.

94. Федеральное агентство по техническому регулированию и метрологии. ГОСТ 53480-2009. Надежность в технике. Термины и определения / Федеральное агентство по техническому регулированию и метрологии. – М. : Стандартинформ, 2010. – 26 с.

95. Беляев, Ю. К. Надежность технических систем: Справочник / Ю. К. Беляев, В. А. Богатырев, В. В. Болотин. – М. : Радио и связь, 1985. – 608 с.

96. Глазунов, Л. П. Основы теории надежности автоматических систем управления / Л. П. Глазунов, Л. П. Грабовецкий, О. В. Щербаков. – Л. : Энергоатомиздат, 1984. – 208 с.

97. Труханов, В. М. Новый подход к обеспечению надежности сложных систем / В. М. Труханов. – М. : Спектр, 2010. – 248 с.

98. Шишмарев, В. Ю. Надежность технических систем / В. Ю. Шишмарев. – М. : Издательский центр «Академия», 2010. – 304 с.

99. Авдуевский, В. С. Надежность и эффективность в технике: Справочник. В 10 т. Т. 5: Проектный анализ надежности / Под ред. В.И.Патрушева и А.И.Рембезы / В. С. Авдуевский. – М. : Машиностроение, 1988. – 316 с.

100. Леонтьев, Е. А. Надежность экономических информационных систем / Е. А. Леонтьев. – Тамбов : Изд-во Тамб. гос. техн. ун-та, 2002. – 128 с.

101. Акимова, Г. П. Методология оценки надежности иерархических информационных систем / Г. П. Акимова, А. В. Соловьев // Труды ИСА РАН. – М., 2006. – Т. 23. – С. 18-47.

102. Липаев, В. В. Надежность программных средств / В. В. Липаев. – М. : Синтег, 1998. – 232 с.

103. Ястребенецкий, М. А. Надежность автоматизированных систем управления технологическими процессами / М. А. Ястребенецкий, Г. М. Иванова. – М. : Энергоатомиздат, 1989. – 264 с.

104. Диллон, Б. Инженерные методы обеспечения надежности систем / Б. Диллон, Ч. Сингх. – М. : Мир, 1984. – 318 с.

105. Дружинин, Г. В. Надежность автоматизированных систем / Г. В. Дружинин. – М. : Энергия, 1977. – 536 с.

106. Ермаков, А. А. Основы надежности информационных систем / А. А. Ермаков. – Иркутск : ИрГУПС, 2006. – 151 с.

107. Чуканов, В. О. Надежность программного обеспечения и аппаратных средств систем передачи данных атомных электростанций / В. О. Чуканов. – М. : МИФИ, 2008. – 168 с.

108. Майерс, Г. Надежность программного обеспечения / Г. Майерс. – М. : Мир, 1980. – 358 с.

109. Шураков, В. В. Надежность программного обеспечения систем обработки данных / В. В. Шураков. – М. : Финансы и статистика, 1987. – 272 с.

110. Тейтер, Т. Надежность программного обеспечения / Т. Тейтер, М. Липов, Э. Нельсон. – М. : Мир, 1981. – 323 с.

111. Grant Ireson, W. Справочник по надежности / W. Grant Ireson, под ред. Левина Б.Р.. – М. : Мир, 1969. – 338 с.

112. ГОСТ 28806-90. Качество программных средств. Термины и определения. [Электронный ресурс]. – 1992. – Режим доступа: http://docs.pravo.ru/document/view/20839227/19928449/.

113. ISO/IEC. ISO/IEC 9126:2001 Information technology - Software engineering — Product quality [Электронный ресурс] / ISO/IEC. – ISO/IEC, 2001. – Режим доступа: http://www.iso.org/iso/catalogue_detail.htm?csnumber=22749 (дата обращения: 15.04.2012).

114. Липаев, В. В. Качество программных средств. Методические рекомендации. / В. В. Липаев. – М. : Янус-К, 2002. – 400 с.

115. Василенко, Н. В. Модели оценки надежности программного обеспечения / Н. В. Василенко, В. А. Макаров // Вестник Новгородского государственного университета. – 2004. – № 28. – С. 126-132.

116. Соколов, С. Н. Количественная оценка надежности деятельности человека-оператора, взаимодействующего с ЭВМ / С. Н. Соколов. – М., 1994. – 135 с.

117. Щебланов, В. Ю. Надежность деятельности и профессиональное здоровье работающих в неблагоприятных условиях / В. Ю. Щебланов. – М., 1996. – 255 с.

118. Бодров, В. А. Психология и надежность: человек в системах управления техникой / В. А. Бодров, В. Я. Орлов. – М. : Институт психологии РАН, 1998. – 288 с.

119. Iwasa, K. WS-Reliability 1.1 [Электронный ресурс] / Kazunori Iwasa // OASIS. – OASIS. – Режим доступа: http://docs.oasis-open.org/wsrm/ws-reliability/v1.1/wsrm-ws_reliability-1.1-spec-os.pdf (дата обращения: 5.04.2009).

120. Papazoglou, M. P. Web Services and Business Transactions / M. P. Papazoglou // World Wide Web:Internet and Web Information Systems. – 2003. – № 6. – С. 49-91.

121. Little, M. Transactions and Web Services / Mark Little // Communications of the ACM. – 2003. – № 10. – С. 49-54.

122. Haugen, B. Multi-Party Electronic Business Transactions [Электронный ресурс] / B. Haugen, T. Fletcher. – ebXML and UN/CEFACT TMG Business Process Work Group, OASIS, 2002. – Режим доступа: http://www.enhyper.com/content/multipartybusinesstransactions.pdf (дата обращения: 05.04.2013).

123. Papazoglou, M. P. Web services technology in support of business transactions [Электронный ресурс] / M. P. Papazoglou, B. Kratz.

124. Роб, П. Системы баз данных: проектирование, реализация и управление / П. Роб, К. Коронел. – СПб. : БХВ-Петербург, 2004. – 1040 с.

125. Конполли, Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика / Томас Конполли, Каролин Бегг. – М. : Издательский дом «Вильямс», 2003. – 1440 с.

126. Саймон, А. Р. Стратегические технологии баз данных: менеджмент на 2000 год / Алан Р. Саймон. – М. : Финансы и статистика, 199. – 479 с.

127. Gray, J. Transaction Processing Concepts and Techniques / Jim Gray. – Morgan Kaufmann, 1993. – 808 с.

128. ISO/IEC 9075-2:2011 [Электронный ресурс] // International Organization for Standardization. – ISO, 2011. – http://jtc1sc32.org/doc/N2151-2200/32N2154T-text_for-ballot-FDIS_9075-2.pdf. – Режим доступа: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=53682 (дата обращения: 5.10.2012).

129. Гарсиа-Молина, Г. Системы баз данных: Полный курс / Гектор Гарсиа-Молина, Джеффри Ульман, Дженнифер Уидом. – М. : Издательский дом «Вильямс», 2003. – 1088 с.

130. Свистунов, А. Н. Построение распределенных систем на Java [Электронный ресурс] / А. Н. Свистунов // Интернет-университет информационных технологий. – Режим доступа: http://www.intuit.ru/department/pl/distrsysjava/14/ (дата обращения: 05.08.2010).

131. Bernstein, P. A. Principles of Transaction Processing / Philip A. Bernstein. – Morgan Kaufmann, 2009. – 382 с.

132. Sun, C. Requirements and Evaluation of Protocols and Tools for Transaction Management in Service Centric Systems / C. Sun, M. Aiello // Материалы конференции «COMPSAC: International Computers, Software & Applications Conference». – Taiwan, 2007. – С. 461-466.

133. Wang, Y. Adaptable Transaction Processing in the Web Services Domain [Электронный ресурс] / Y. Wang, Y. Weihai. – Режим доступа: http://arno.uvt.nl/show.cgi?fid=69154 (дата обращения: 12.05.2010).

134. Schmit, B. Towards Transactional Web Services / B. Schmit // Материалы конференции «Seventh IEEE International Conference on E-Commerce Technology Workshops». – 2005. – С. 12-20.

135. Doreen, T. An analysis of transactions in service-centric systems [Электронный ресурс] / T. Doreen // Department of Mathematics and Computing Science Rijksuniversiteit Groningen. – 2010. – Режим доступа: http://www.cs.rug.nl/~aiellom/tesi/tuheirwe.pdf (дата обращения: 15.02.2013).

136. Alonso, G. Advanced Transaction Models in Workflow Contexts / Gustavo Alonso, Agrawal Divyakant, Amr El Abbadi, Mohan Kamath, Roger Günthör, C. Mohan // Материалы конференции «ICDE '96 Proceedings of the Twelfth International Conference on Data Engineering». – 1996. – С. 574-581.

137. Prochazka, M. Advanced Transactions in Component-Based Software Architectures [Электронный ресурс] / M. Prochazka.

138. Gray, J. The transaction concept: virtues and limitations / Jim Gray // Материалы конференции «Seventh international conference on Very Large Data Bases». – 1981. – С. 144-154.

139. Tang, F. A pipeline-based approach for long transaction processing in web service environments [Электронный ресурс] / F. Tang, I. You, L. Li, C. Wang, Z. Cheng, S. Guo.

140. Tang, F. Automatic transaction compensating for reliable grid applications / F. Tang, M. Li, J. Huang // Journal of Computer Science and Technology. – 2006. – № 4. – С. 529-536.

141. Aikebaier, A. A protocol for reliably, flexibly, and efficiently making agreement among peers / A. Aikebaier, M. Takizawa // International Journal of Web and Grid Services. – 2009. – № 4. – С. 356-371.

142. Dalal, S. Coordinating business transactions on the web / S. Dalal, S. Temel, M. Little, M. Potts, J. Webber // IEEE Internet Computing. – 2003. – № 1. – С. 30-39.

143. Potts, M. Business Transaction Protocol Primer [Электронный ресурс] / M. Potts, B. Cox, B. Pope. – OASIS, 2002. – Режим доступа: http://www.oasis-open.org/committees/business-transaction/documents/primer/Primerhtml/BTP%20Primer%20D1%2020020602.html (дата обращения: 05.04.2013).

144. Limthanmaphon, B. Web service composition transaction management / Benchaphon Limthanmaphon, Yanchun Zhang // ADC '04 Proceedings of the 15th Australasian database conference. – 2004. – Т. 27. – С. 171-179.

145. Michael, B. Precise Modelling of Compensating Business Transactions and its Application to BPEL / Butler Michael, Carla Ferreira, Muan Yong Ng // Journal of Universal Computer Science. – 2004. – Т. 11, № 5. – С. 712-743.

146. Волкова, В. Н. Системный анализ и принятие решений: Словарь-справочник / В. Н. Волкова, В. Н. Козлов. – М. : Высш.шк., 2004. – 616 с.

147. Новосельцев, В. И. Теоретическиие основы системного анализа / В. И. Новосельцев. – М. : Майор, 2006. – 592 с.

148. Садовский, В. Н. Основания общей теории систем / В. Н. Садовский. – М. : Наука, 1974. – 280 с.

149. Прангишвили, И. В. Системный подход и общесистемные закономерности. Серия «Системы и проблемы управления» / И. В. Прангишвили. – М. : СИНТЕГ, 2000. – 528 с.

150. Артамонов, И. В. Оркестровка и хореография: подходы к описанию композитных бизнес-процессов / И. В. Артамонов // Применение математических методов и информационных технологий в экономике. – 2011. – № 10. – С. 65-75.

151. Нечипоренко, В. И. Структурный анализ систем / В. И. Нечипоренко. – М. : Советское радио, 1977. – 216 с.

152. Черняк, Ю. И. Системный анализ в управлении экономикой / Ю. И. Черняк. – М. : Экономика, 1975. – 191 с.

153. Бусленко, Н. П. Моделирование сложных систем / Н. П. Бусленко. – М. : Наука, 1968. – 356 с.

154. Поцелуев, А. В. Статистический анализ и синтез сложных динамических систем / А. В. Поцелуев. – М. : Машиностроение, 1984. – 208 с.

155. Кобелев, Н. Б. Основы имитационного моделирования сложных экономических систем / Н. Б. Кобелев. – М. : Дело, 2003. – 336 с.

156. Колесов, Ю. Б. Объектно-ориентированное моделирование сложных динамических систем / Ю. Б. Колесов. – Санкт-Петербург : Издательство СПбГПУ, 2004. – 232 с.

157. Шеннон, Р. Имитационное моделирование систем - искусство и наука / Роберт Шеннон. – М. : Мир, 1978. – 420 с.

158. Волкова, В. Н. Теория систем и системный анализ в управлении организациями / В. Н. Волкова, А. А. Емельянов. – М. : Финансы и статистика, 2006. – 848 с.

159. Амбросов, В. Н. Системный анализ в управлении социально-экономической системой: учебно-методическое пособие. / В. Н. Амбросов. – Иркутск : ИГЭА, 2001. – 71 с.

160. Капур, К. Надежность и проектирование систем / К. Капур, Л. Ламберсон. – М. : Мир, 1980. – 610 с.

161. Дорохов, А. Н. Обеспечение надежности сложных систем / А. Н. Дорохов, А. Н. Миронов, В. А. Керножицкий, О. Л. Шестопалова. – СПб. : Лань, 2011. – 352 с.

162. Ллойд, Д. Надежность: организация исследования, методы, математический аппарат / Д. Ллойд, М. Липов. – М. : Советское радио, 1964. – 687 с.

163. Антонов, А. В. Системный анализ / А. В. Антонов. – М. : Высш. шк., 2004. – 454 с.

164. Рябинин, И. А. Логико-вероятностные методы исследования надежности структурно-сложных систем / И. А. Рябинин, Г. Н. Черкесов. – М. : Радио и связь, 1981. – 264 с.

165. Советов, Б. Я. Моделирование систем: учеб. для вузов / Б. Я. Советов, С. А. Яковлев. – М. : Высш. шк., 2001. – 343 с.

166. Алексеев, А. Е. Диагностика надежности автоматизированных систем / А. Е. Алексеев. – Архангельск : ГОУ АГТУ, 2004. – 75 с.

167. Айвазян, С. А. Прикладная статистика и основы эконометрики / С. А. Айвазян, В. С. Мхитарян. – М. : Юнити, 1998. – 1022 с.

168. Sun, H. CoBTx-Net: A Model for Reliability Verification of Collaborative Business Transaction / Haiyang Sun, Jian Yang // Business Process Management

Workshops / Arthur ter Hofstede, Boualem Benatallah, Hye-Young Paik. – Brisbane : Springer, 2007. – С. 220-231.

169. Артамонов, И. В. Надежность бизнес-транзакций в сервис-ориентированной среде / И. В. Артамонов // Материалы конференции «Инновационные информационные технологии». – Прага, 2013. – С. 12-19.

170. Артамонов, И. В. Надежность B2B-взаимодействия в сервис-ориентированной среде / И. В. Артамонов // APRIORI. Серия: Естественные и технические науки. – Краснодар : APRIORI, 2014. – № 5. – С. 1-13.

171. Papazoglou, M. P. A Business-Aware Web Services Transaction Model / Mike P. Papazoglou, Kratz Benedikt // Материалы конференции «ICSOC'06 Proceedings of the 4th international conference on Service-Oriented Computing». – 2006. – С. 352-364.

172. Wang, T. Flexible Business Transaction Composition in Service-Oriented Environments / Ting Wang, Jochem Vonk, Paul Grefen. – Eindhoven : Technische Universiteit Eindhoven, 2005. – 29 с.

173. Grefen, P. Global transaction support for workflow management systems: from formal specification to practical implementation / Paul Grefen, Jochem Vonk, Peter Apers // The VLDB Journal — The International Journal on Very Large Data Bases. – 2001. – Т. 10, № 4. – С. 316-333.

174. Grefen, P. Transactional Workflows or Workflow Transactions? / Paul W. P. J. Grefen // DEXA '02 Proceedings of the 13th International Conference on Database and Expert Systems Applications. – 2002. – №2. – С. 60-69.

175. Liu, C. Confirmation: increasing resource availability for transactional workflows / Chengfei Liu, Xuemin Lin, Maria Orlowska, Xiaofang Zhou // Information Sciences: an International Journal. – 2003. – Т. 153, № 1. – С. 37-53.

176. Alonso, G. Workflow Management: The Next Generation of Distributed Processing Tools / G. Alonso, C. Mohan // Advanced Transaction Models and Architectures / Sushil Jajodia, Larry Kerschberg. – Norwell : Kluwer Academic, 1997. – С. 34-56.

177. Devashish, W. Transactions in Transactional Workflows / Worah Devashish, Amith Sheth // Advanced Transaction Models and Architectures / Sushil Jajodia, Larry Kerschberg. – Kluwer Academic, 1997. – С. 3-33.

178. Medjahed, B. Semantic Web Enabled Composition of Web Services / Brahim Medjahed. – Falls Church : Faculty of the Virginia Polytechnic Institute, 2004. – 263 с.

179. Leutenmayr, S. Selected Languages for Web Services Composition: Survey, Challenges, Outlook / Stephan Leutenmayr. – 2007. – 109 с.

180. Segev, A. Context-Based Matching and Ranking of Web Services for Composition / Aviv Segev, Eran Toch // IEEE Transactions on Services Computing. – 2009. – Т. 2, № 3. – С. 210-222.

181. Khadka, R. An Evaluation of Dynamic Web Service Composition Approaches / Ravi Khadka, Brahmananda Sapkota // Материалы конференции «4th International Workshop on Architectures, Concepts and Technologies for Service Oriented Computing». – 2010. – С. 67-79.

182. Pat Pik Wah Chan, Dynamic Web Service Composition: A New Approach in Building Reliable Web Service / Pat Pik Wah Chan, Michael R. Lyu // Материалы конференции «Proceedings of the 22nd International Conference on Advanced Information Networking and Applications». – 2008. – С. 20-25.

183. Baryannis, G. Automated Web Service Composition: State of the Art and Research Challenges / George Baryannis, Dimitris Plexousakis. – 2010. – 82 с.

184. ANR PERSO. State of the art: Models and Algorithms for Service Composition [Электронный ресурс] / ANR PERSO. – 2009. – Режим доступа: http://anr-perso.ibisc.univ-evry.fr/PublicDocuments/ (дата обращения: 02.04.2012).

185. Kapitsaki, G. Mobile and Wireless Communications Summit, 2007. 16th IST / G. Kapitsaki, D.A. Kateros, I.E. Foukarakis, G. N. Prezerakos, D.I. Kaklamani, I.S. Venieris // Материалы конференции «Service Composition: State of the art and future challenges». – 2007. – С. 1-5.

186. Rao, J. A Survey of Automated Web Composition Methods / Jinghai Rao, Xiaomeng Su // In Proceedings of the First International Workshop on Semantic Web Services and Web Process Composition. – 2004. – № 2. – С. 43-54.

187. Xia, J. QoS-based Service Composition / Jinchun Xia // Computer Software and Applications Conference, 2006. COMPSAC '06. 30th Annual International. – 359 - 361. – Т. 2. – С. 2006.

188. Bocchi, L. Transactional Aspects in Coordination and Composition of Web Services (Ph.D. Thesis) / Laura Bocchi. – University of Bologna, 2006. – 118 с.

189. Bhiri, S. Transactional Patterns for Reliable Web Services Compositions / Sami Bhiri, Oliver Perrin, Claude Godart // Материалы конференции «ICWE '06 Proceedings of the 6th international conference on Web engineering». – 2006. – С. 137-144.

190. Fauvet, M. Handling Transactional Properties in Web Service Composition / Marie-Christine Fauvet, Helga Duarte, Marlon Dumas, Boualem Benatallah // Ma-

териалы конференции «WISE'05 Proceedings of the 6th international conference on Web Information Systems Engineering». – 2005. – С. 273-289.

191. Guceglioglou, A. PQMM: A new model to measure business process quality / A. Selcuk Guceglioglou, Onur Demirors // BPM and Workflow Handbook / Layna Fischer. – Future Strategies, 2008. – С. 139.

192. Khoshafian, S. BPM Center of Excellence Manifesto / Setrag Khoshafian // 2007 BPM and Workflow Handbook / Layna Fischer. – Future Strategies Inc., 2007. – С. 73-84.

193. Cardoso, J. Modeling Quality of Service for Workflows and Web Service Processes / Jorge Cardoso, John Miller, Amit Sheth, Jonathan Arnold // Journal of Web Semantics. – 2004. – № 1. – С. 281-308.

194. Международный союз электросвязи. E.800 Определение терминов, относящихся к качеству обслуживания / Международный союз электросвязи. – 2008. – 32 с.

195. Jin, L. Analysis on Service Level Agreement of Web Services / HP Laboratories. – 2002. – 13 с.

196. Arsanjani, A. Web Services: Promises and Compromises / IBM Research Division. – 2002. – 19 с.

197. Tsai, W. A software reliability model for Web services / W. T. Tsai, D. Zhang, Y. Chen, H. Huang, R. Paul, N. Liao // Материалы конференции «The 8th IASTED International Conference on Software Engineering and Applications». – 2004. – С. 144-149.

198. Canfora, G. QoS-Aware Replanning of Composite Web Services / Gerardo Canfora, Massimiliano Di Penta, Raffaele Esposito, Maria Luisa Villani // Материалы конференции «ICWS '05 Proceedings of the IEEE International Conference on Web Services». – 2005. – С. 121-129.

199. Chandrasekaran, S. 'Performance Analysis and Simulation of Composite Web Services. / Senthilanand Chandrasekaran, John Miller, G. Silver, I. Arpinar, Amit P. Sheth // Electronic Markets. – 2003. – Т. 13, № 2. – С. 120-132.

200. Yu, T. The Design of QoS Broker Algorithms for QoS-Capable Web Services / Tao Yu, Kwei-Jay Lin // Материалы конференции «EEE '04 Proceedings of the 2004 IEEE International Conference on e-Technology, e-Commerce and e-Service (EEE'04)». – 2004. – С. 17-24.

201. Zeng, L. QoS-A ware Middleware for Web Services Composition / Liangzhao Zeng, Boualem Benatallah, Anne H.H. Ngu, Marlon Dumas, Jayant

Kalagnanam, Henry Chang // IEEE Transactions on Software Engineering. – 2004. – Т. 30, № 5. – С. 311-327.

202. Aiello, M. Optimal QoS-Aware Web Service Composition / Marco Aiello, Elie el Khoury, Alexander Lazovik, Patrick Ratelband // Материалы конференции «IEEE Conference on Commerce and Enterprise Computing, 2009. CEC '09». – 2009. – С. 491-494.

203. Goel, A. Software Reliability Models: Assumptions, Limitations, and Applicability / Amrit L. Goel // IEEE Transactions on Software Engineering. – 1985. – Т. 11, № 12. – С. 1411-1423.

204. Zhang, J. Criteria Analysis and Validation of the Reliability of Web Services-oriented Systems / Jia Zhang, Liang-Jie Zhang // ICWS '05 Proceedings of the IEEE International Conference on Web Services. – 2005. – № 5. – С. 621-628.

205. van der Aalst, W. Workflow management models, methods, and systems / Wil van der Aalst, Kees Max van Hee. – Academic Service, 1997. – 368 с.

206. Samaranayake, S. Fault-Tolerant Business Processes [Электронный ресурс] / Sanka Samaranayake, Ricardo Jiménez-peris, Marta Patiño-martínez // Facultad de Informatica Universidad Politecnica de Madrid. – 2011. – Режим доступа: http://lsd.ls.fi.upm.es/lsd/papers/2011/Fault_Tolerant_Business_Processes.pdf (дата обращения: 17.10.2013).

207. Cardoso, J. Workflow Quality of Service Management using Data Mining Techniques / Jorge Cardoso // Intelligent Systems, 2006 3rd International IEEE Conference. – 2006. – № 3. – С. 479-482.

208. Cardoso, J. Workflow Quality of Service / LSDIS Lab, Computer Science ; Univiersity of Georgia. – Athens GA USA, 2002. – 13 с.

209. Zhai, Y. SOA Middleware Support for Service Process Reconfiguration with End-to-End / Yanlong Zhai, Jing Zhang, Kwei-Jay Lin // Материалы конференции «IEEE International Conference on Web Services». – 2009. – С. 815-822.

210. Артамонов, И. В. Внутренне-надежная бизнес-транзакция / И. В. Артамонов // Материалы конференции «Инновации на основе информационных и коммуникационных технологий». – М. : Московский институт электроники и математики НИУ ВШЭ, 2013. – С. 510-512.

211. Zheng, Z. Optimal Fault Tolerance Strategy Selection for Web Services / Zibin Zheng, Michael R. Lyu // International Journal of Web Services Research. – 2010. – Т. 7. – С. 21-40.

212. Zo, H. Measuring Reliability of Applications Composed of Web Services / Hangjung Zo, Derek L. Nazareth, Hemant K. Jain // Материалы конференции «Pro-

ceedings of the 40th Hawaii International Conference on System Sciences». – 2007. – С. 278.

213. Барлоу, Р. Статистическая теория надежности и испытания на безотказность. Пер. с англ. / Р. Барлоу, Ф. Прошан. – М. : Наука, 1984. – 328 с.

214. Райншке, К. Модели надежности и чувствительности систем / К. Райншке. – М. : Мир, 1979. – 453 с.

215. Райншке, К. Оценка надежности систем с использованием графов / К. Райншке, И. А. Ушаков. – М. : Радио и связь, 1988. – 208 с.

216. Jensen, K. Coloured Petri Nets and CPN Tools for modelling and validation of concurrent systems / Kurt Jensen, Lars Michael Kristensen, Lisa Wells // International Journal on Software Tools for Technology Transfer (STTT). – 2007. – Т. 9, № 3. – С. 213-254.

217. Артамонов, И. В. Моделирование сервис-ориентированной архитектуры с помощью окрашенных сетей Петри / И. В. Артамонов // Вестник Новосибирского государственного университета. Серия: Информационные технологии. – Новосибирск : Новосибирский национальный исследовательский государственный университет, 2014. – Т. 12, № 2. – С. 5-13.

218. Ньюкомер, Э. Веб-сервисы: XML, WSDL, SOAP и UDDI / Эрик Ньюкомер. – СПб. : Питер, 2003. – 256 с.

219. Артамонов, И. В. Моделирование сервисной композиции с помощью окрашенных сетей Петри / И. В. Артамонов // Вестник НГУЭУ. – Новосибирск, 2013. – № 2. – С. 180-187.

220. Артамонов, И. В. Моделирование надежных B2B-взаимодействий с помощью окрашенных сетей Петри / И. В. Артамонов // Труды международного симпозиума «Надежность и качество». – Пенза, 2013. – Т. 1, № 1. – С. 146-148.

221. Artamonov, I. V. Coloured Petri Net usage for modeling a business transaction in a service-oriented environment / I. V. Artamonov // Материалы конференции «Innovative Information Technologies: Materials of the International scientific - practical conference». – Prague, 2014. – С. 397-402.

222. Артамонов, И. В. Использование окрашенных сетей Петри для моделирования бизнес-транзакций в сервис-ориентированной среде / И. В. Артамонов // Известия Иркутской государственной экономической академии (Байкальский государственный университет экономики и права). – 2013. – № 5. – С. 25.

223. van der Aalst, W. Verification of Workflow Nets / W.M.P. van der Aalst // Материалы конференции «Proceedings of the 18th International Conference on Application and Theory of Petri Nets». – 1997. – С. 407-426.

224. Гихман, И. И. Теория вероятностей и математическая статистика / И. И. Гихман, А. В. Скороход, М. И. Ядренко. – Киев : Вища школа, 1979. – 408 с.

225. Артамонов, И. В. Анализ устойчивости бизнес-транзакций с помощью цепей Маркова / И. В. Артамонов // Информационные системы и технологии. – 2015. – Т. 3, № 89. – С. 41-46.

226. Соколов, Г. А. Теория вероятностей. Управляемый цепи Маркова в экономике / Г. А. Соколов, Н. А. Чистякова. – М. : ФИЗМАТЛИТ, 2005. – 248 с.

227. Питерсон, Д. Теория сетей Петри и моделирование систем / Дж. Питерсон. – М. : Мир, 1984. – 264 с.

228. Артамонов, И. В. Исследование надежности работы интернет-магазина с помощью окрашенных сетей Петри / И. В. Артамонов // Известия Волгоградского государственного технического университета. – 2014. – № 22. – С. 42-47.

229. Саломатин, Н. А. Имитационное моделирование в оперативном управлении производством / Н. А. Саломатин, В. Ф. Петроченко, Г. В. Беляев, Е. В. Прошлякова. – М. : Машиностроение, 1984. – 208 с.

230. Артамонов, И. В. Программный комплекс анализа надежности бизнес-транзакции / И. В. Артамонов // Информационные системы и технологии. – Орел : Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Государственный университет - учебно-научно-производственный комплекс», 2014. – № 5 (85). – С. 5-13.

231. Software I Maintain [Электронный ресурс] // Michael Westergaard. – Michael Westergaard, 2013. – Режим доступа: https://westergaard.eu/2012/06/software-i-maintain/ (дата обращения: 06.05.2013).

232. CPN Tools 4.0 [Электронный ресурс]. – 2013. – Режим доступа: http://cpntools.org/start (дата обращения: 05.06.2013).

233. Артамонов, И. В. Свободное программное обеспечение: преимущества и недостатки / И. В. Артамонов // Известия ИГЭА. – 2012. – № 5. – С. 122-125.

234. Суходолов, А. П. Современные информационно-телекоммуникационные технологии в управлении социально-экономическими системами / А. П. Суходолов, Т. Г. Озерникова, В. В. Братищенко, З. В. Архипова, Д. И. Сачков, А. В. Родионов, И. В. Артамонов, И. А. Кузнецова, Д. С. Матусевич. – Иркутск : БГУЭП, 2013. – 196 с.

235. Красовский, А. А. Справочник по теории автоматического управления / А. А. Красовский. – М. : Наука, 1987. – 711 с.

236. Таха, Х. Введение в исследование операций / Х. Таха. – М. : Издательский дом «Вильямс», 2001. – 912 с.

237. Горяинов, В. Б. Математическая статистика / В. Б. Горяинов, И. В. Павлов, Г. М. Цветкова. – М. : Изд-во МГТУ им. Н.Э.Баумана, 2001. – 424 с.

238. Кобзарь, А. И. Прикладная математическая статистика / А. И. Кобзарь. – М. : ФИЗМАТЛИТ, 2006. – 816 с.

239. Клейнен, Д. Статистические методы в имитационном моделировании. Т. 2 / Дж. Клейнен. – М. : Статистика, 1978. – 335 с.

240. Большев, Л. Н. Таблицы математической статистики / Л. Н. Большев, Н. В. Смирнов. – М. : Наука, 1983. – 416 с.

241. ван дер Варден, Б. Математическая статистика / Б.Л. ван дер Варден. – М. : Издательство иностранной литературы, 1960. – 435 с.

242. Лемешко, Б. Ю. Статистический анализ данных, моделирование и исследование вероятностных закономерностей. Компьютерный подход: монография / Б. Ю. Лемешко, С. Б. Лемешко, С. Н. Постовалов, Е. В. Чимитова. – Новосибирск : Издательство НГТУ, 2011. – 888 с.

243. Барлоу, Р. Математическая теория надежности / Под ред. Гнеденко, Б. В. / Р. Барлоу, Ф. Прошан. – М. : Советское радио, 1969. – 488 с.

244. Хенли, Э. Надежность технических систем и оценка риска / Э. Дж Хенли, Х. Кумамото. – М. : Машинстроение, 1984. – 528 с.

Приложение А. Пошаговое описание работы с программой

Опишем порядок работы с ПКБТ на примере схемы сети Петри, описывающей простую бизнес-транзакцию.

Пусть дана бизнес-транзакция, представляющая бизнес-процесс с одним основным участником – некоторым оператором (человеком, функцией, сервисом – не имеет значения). Задача оператора – принимать входящие значения и увеличивать их на единицу, т.е. преобразовывать вход процесса в выход путем увеличения значения входных данных на 1. Между оператором и входом/выходом бизнес-процесса существуют некоторые промежуточные позиции хранения результатов.

Положим, что необходимо исследовать надежность этой бизнес-транзакции при условии, что информация о надежности функционирования оператора известна, а остальные участники бизнес-транзакции предполагаются абсолютно надежными. Это необязательное условие, введенное в целях упрощения примера, т.к. ПКБТ позволяет оценивать надежность при неограниченном количестве ненадежных участников.

Положим также, что бизнес-транзакция функционирует в сервис-ориентированной среде, что позволяет нам говорить о возможности применения сервисов, которые либо дублируют оператора, либо проводят отмена выполнения бизнес-транзакции. Т.е. на схему мы можем добавить также компенсирующие и восстанавливающие сети.

Отрисовка схемы бизнес-транзакции

Окно среды отрисовки схем представляет собой три области (см. рис. 26): левая область с меню и информацией о текущей схеме, внутреннее окно со схемой и окно с инструментами отрисовки. Окон инструментов отрисовки и окон схем может быть несколько, но в данном случае они показаны по одному.

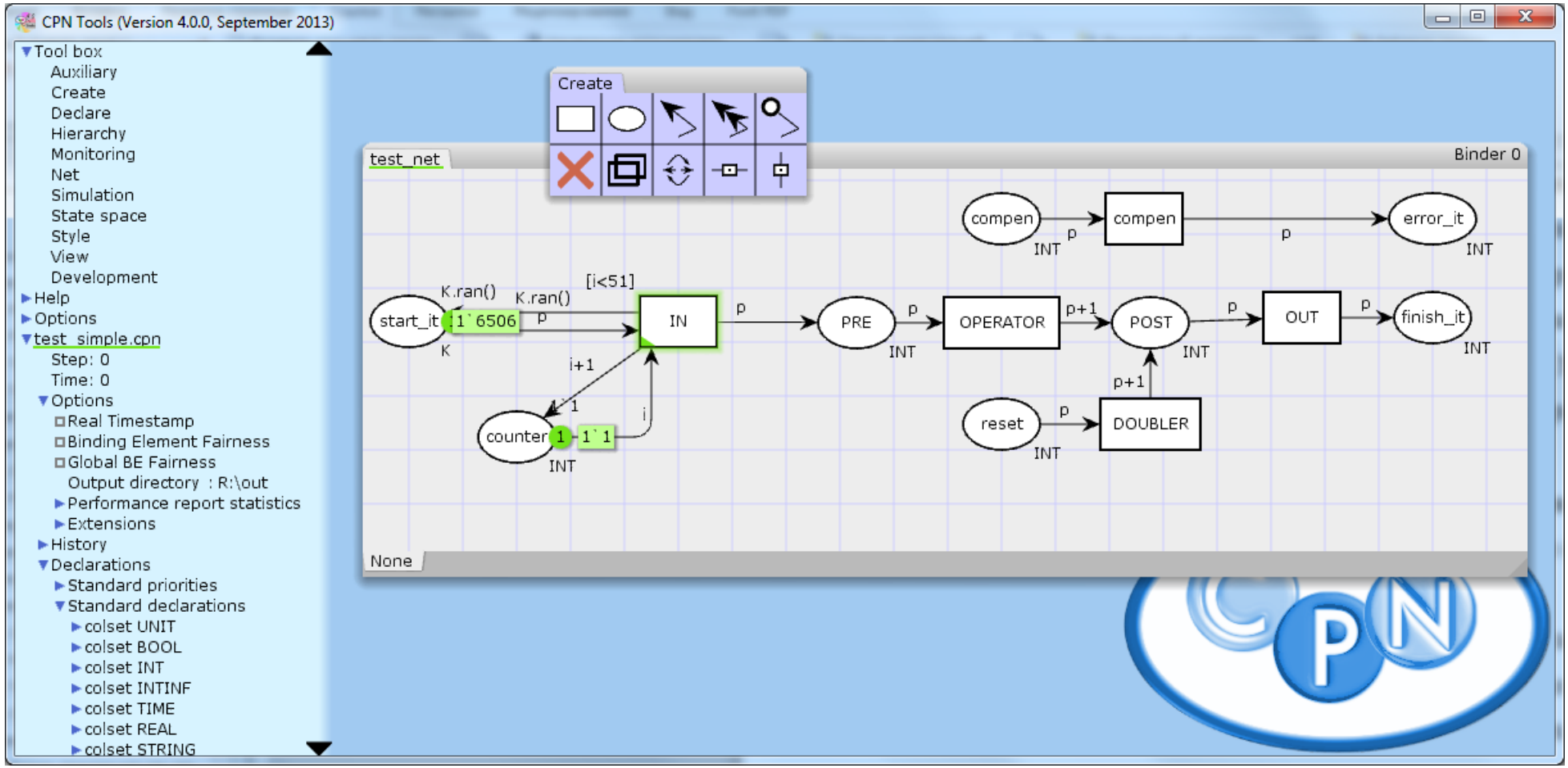


Рисунок 26 – Окно среды отрисовки схем

Схема бизнес-транзакции состоит следующих элементов, показанных на рис. 27.

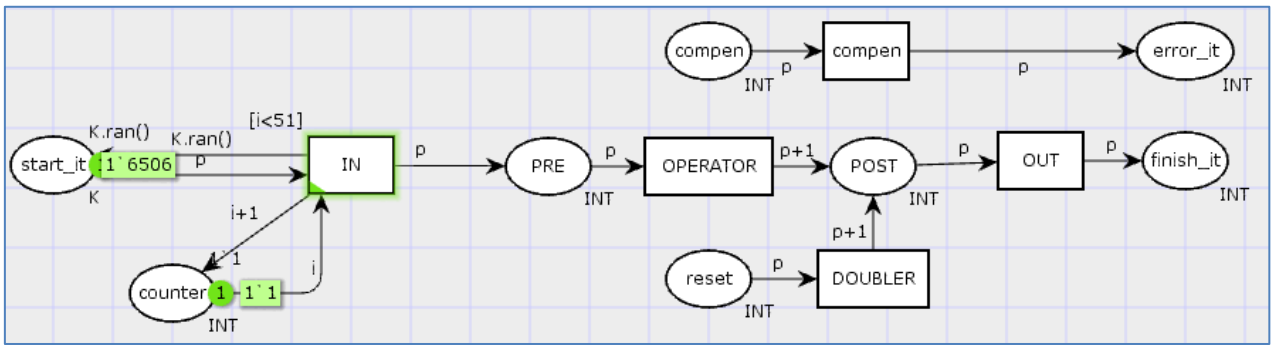


Рисунок 27 – Схема бизнес-транзакции

- Позиция start_it – позиция, содержащая входные данные для бизнес-транзакции. В целях моделирования данные представлены случайными числами от 0 до 10000.
- Позиция counter – позиция, позволяющая вести отсчет количества введенных элементов. В данном случае количество входных данных ограничено 50 элементами.
- Позиция PRE содержит данные, поступающие на вход оператора.
- Позиция POST содержит данные, поступающие от оператора.
- Позиция finish_it содержит данные, поступающие на выход бизнес-процесса.
- Переход IN передает данные оператору при условии, что их было меньше 50 (для этого у него определен специальный страж).
- Переход OPERATOR выполняет функцию оператора – увеличивает элемент на 1.
- Переход OUT передает информацию от оператора на выход.

Позиции compen и error_it входят в компенсирующую сеть. Позиция compen принимает данные в случае сбоя, переход compen их обрабатывает (в данном случае – просто передает далее), а позиция error_it хранит информацию обо всех сбоях.

Позиция reset и переход DOUBLER составляют восстанавливающую сеть. Позиция reset принимает данные в случае сбоя, переход их обрабатывает по методу, аналогичному методу перехода OPERATOR. Иными словами, эта восстанавливающая сеть демонстрирует поведение обычного резервного элемента.

Отметим, что восстанавливающие и компенсирующие сети не предусмотрены условием задачи, но введены нами намеренно для демонстрации возможностей ПКБТ. В первой итерации анализа надежности эти вспомогательные сети будут игнорироваться.

Инициализация эксперимента

Основной интерфейс ПКБТ представлен областью меню с логотипом в верхней части окна, областью главной информации в центре и областью авторских прав в нижней части экрана. В главном окне (на главной странице) ПКБТ представлена краткая информация, требования к схемам и приглашение ознакомится с основными научными статьями и публикациями автора на сайте научной библиотеки.

Чтобы начать работу по оценке надежности, необходимо перейти в окно «Новый эксперимент» и следовать пошаговым инструкциям подготовки эксперимента. На первом этапе (см. рис. 28) необходимо загрузить в ПКБТ созданную ранее схему процесса в виде файла с расширением «*.cpn» и определить основные настройки эксперимента.

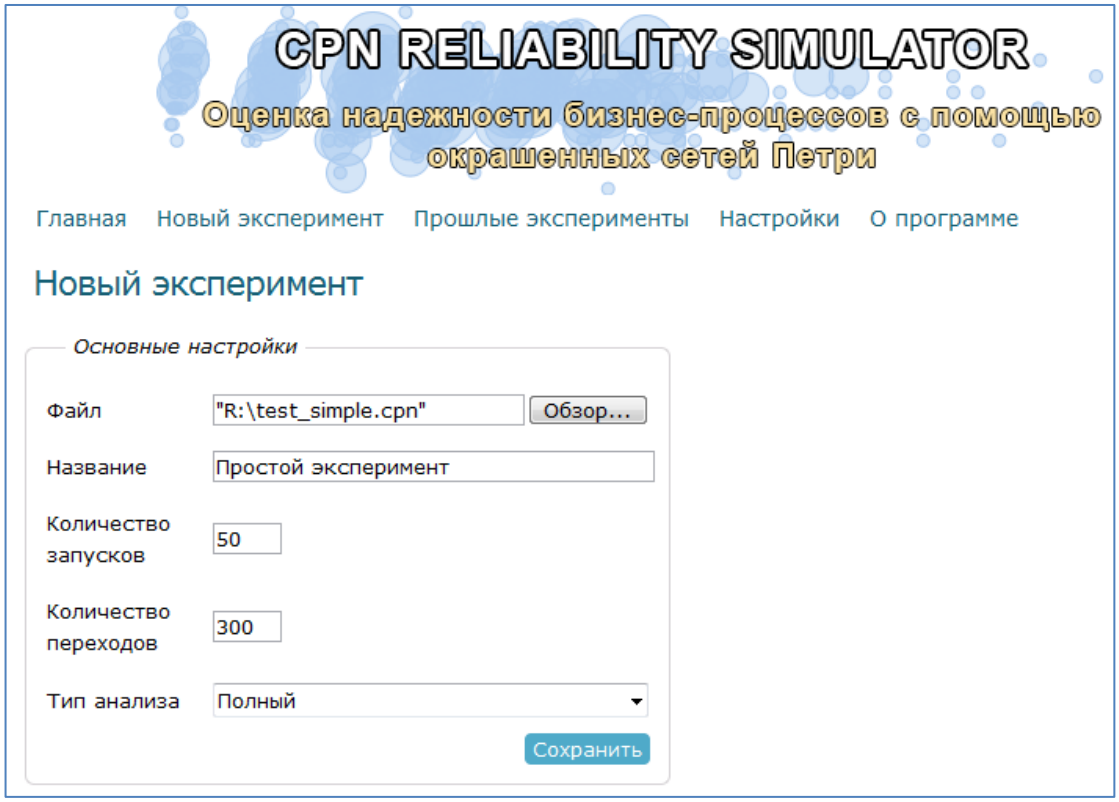


Рисунок 28 – Начало эксперимента

К основным настройками относятся:

- Название эксперимента.
- Точное количество порций исходных данных в схеме процесса (количество запусков процесса). Это условие необходимо для расчета количества успешных и неуспешных запусков процесса при одной имитации. Такое количество запусков процесса проведет ПКБТ во время одного прогона имитационного эксперимента. Определяя в примере значение

«50», мы предполагаем, что на вход бизнес-транзакции последовательно поступает 50 заявок, которые она должна обработать. Эти заявки будут одновременно активны, что позволит оценить работу системы в условиях параллельной обработки множества запросов. Обработка всех 50 заявок и будет считаться одним прогоном, результаты которого подвергаются оценке.

- Приблизительное количество переходов внутри схемы при каждом запуске от начала процесса до его завершения – успешного или нет. Это условие необходимо для того, чтобы ПКБТ смог прервать процесс имитации очередного запуска процесса по достижении этого показателя. Прерывание важно, так как в схеме могут быть внутренние бесконечные циклы, счетчики и переходы. Если никаких «бесконечных» процессов в схеме нет, то ее имитация остановится естественным образом – при достижении финальной маркировки. В нашем примере, на обработку 50 заявок требуется 300 переходов.

Выбор типа анализа надежности бизнес-транзакции

В окне настроек нового эксперимента присутствует поле «Тип анализа надежности», которое предполагает выбор: анализ в целом, анализ чувствительности к отдельным операциям, полный анализ и анализ устойчивости. Первые два типа в целом соответствуют задачам № 1 и № 3 оценки надежности (см. главу 1.3.6). Задачи № 2 и № 4 в ПКБТ предполагается решать итерационно, последовательным запуском нескольких экспериментов. Третий тип – «полный анализ» предполагает последовательное выполнение анализа в целом и анализа чувствительности и совместный вывод результатов. Анализ устойчивости реализуется предложенный в 2.7 алгоритм анализа устойчивости с помощью цепей Маркова.

Анализ чувствительности, проводимый ПКБТ, в своей основе имеет ту же цель, что и методы анализа чувствительности в теории автоматического управления [235], исследования операций [236] или имитационного моделирования [157]: изучение изменения поведения модели в ответ на изменение входных параметров. Однако задачи, решаемые ПКБТ применительно к бизнес-транзакциям, отличаются от этих подходов. ПКБТ изучает не характер изменчивости модели, чтобы выяснить допустимые пределы ее адекватности, а то, как бизнес-транзакция меняет свое поведение в зависимости от изменения входных параметров. Другими словами, анализ чувствительности здесь является частью

процесса имитационного моделирования, а не построения модели, как в других дисциплинах.

Как уже отмечалось в 1.3.6, аналитически чувствительность модели может задаваться, например, некоторым регрессионным уравнением, где выходной параметр представлен зависимой переменной, а коэффициенты при независимых задают степень их влияния. Однако, исходя из того, что сложность бизнес-транзакции зачастую не позволяет получить описание ее поведения в таком упрощенном виде, а рассмотрение корреляционных отношений между параметрами не имеет смысла, так как подсистемы бизнес-транзакции могут реализовывать сложные, нелинейные алгоритмы преобразования входных данных. Поэтому в ПКБТ результаты анализа чувствительности представлены в виде корреляционных полей [237] – двумерных графиков, где по каждой оси располагаются значения двух сравниваемых переменных. Интерпретация полей входит в задачу исследователя, и, на наш взгляд, не является трудной задачей: графики цветные, наглядные, поддерживающие масштабируемость и показывающие соотношения между зависимым и независимым параметров «на лету». Для облегчения работы эксперта ПКБТ сравнивает приращение вектора параметра с приращением вектора отклика модели и дает свои рекомендации в подписи графика.

Анализ чувствительности происходит по следующему алгоритму: измеряется связь между каждым исследуемым фактором и откликом системы, причем остальные факторы считаются неизменными – стабильными (с неизменяемым уровнем надежности) или абсолютно надежными. То есть имеет место эксперимент вида «один фактор в каждый момент времени». По умолчанию каждый фактор исследуются на 3-х уровнях надежности, однако эту настройку можно изменить. На наш взгляд, применение этого метода в отличие от факторного эксперимента, где комбинируются все уровни одного фактора со всеми уровнями другого, больше подходит для автоматизированного анализа надежности. Это связано с тем, что:

- Каждый исследуемый подпроцесс может либо выполнить свою операцию, либо отказать в ее выполнении. То есть существует событие отказа, которое может наступить или не наступить при выполнении операции. С одной стороны, наступление событие легко отобразить в полном факторном эксперименте типа 2^k, где 2 – число уровней факторов, а k – количество факторов. Такой план хорошо подойдет, например, для построения линейной регрессионной модели. Отклик также бы измерялся двумя уровнями. Но в бизнес-транзакциях, как и вообще в бизнес-процессах, часто встречаются линейные, последовательные схемы со-

единения элементов, и отказ любого предыдущего элемента вызовет неработоспособность всех последующих и весьма вероятно отказ всей транзакции. Поэтому ввиду такого смешения и влияния одних факторов на другие, использование эксперимента типа 2^k не кажется нам целесообразным. Тем не менее, эксперимент типа 2^k, позволит, например, достоверно выявлять критические для функционирования бизнес-транзакции элементы, и этот метод может быть реализован в ПКБТ при дальнейшем его развитии.

- Применение техники регрессионного анализа для повышения его эффективности требует ручного труда и специализированных знаний, что повышает требования к исследователю и тем самым сильно сужает область применения ПКБТ. При этом в реальности отказ элемента может быть определен некоторой вероятностью, и исследование влияния отказов различной вероятности одних факторов на другие еще более усложняет процесс анализа. Сложность применения регрессионного анализа не позволяет также говорить об оптимальности плана эксперимента [238].
- Полные факторные планы, даже 2^k, требуют большого количества опытов, например, 2^7 требует 128 прогонов. Исходя из того, что каждый прогон может выполняться несколько минут и требовать для своего выполнения вызов сторонних программ, выполнение такого плана потребует большого количества времени. Эффективное использование дробных планов требует соответствующей квалификации эксперта и ручных вычислительных операций.

Описание сетей и определение надежности операций

Между первым и вторым шагом ПКБТ проводит анализ загруженной схемы. Он ищет все сети и их составляющие: арки, позиции и переходы.

На втором шаге подготовки эксперимента необходимо обозначить характер анализируемых сетей, т.е. указать для каждой:

- Название и описание.
- Тип – главная подсеть или второстепенная. Может существовать только одна главная анализируемая сеть на схеме.
- Финишная позиция подсети – позиция, достижение которой будет засчитываться за успешное прохождение сети.

Также на втором шаге необходимо указать, какие подпроцессы (переходы) подвергаются анализу, а какие принимают абсолютно надежными. Для каждого

анализируемого процесса можно указать закон распределения вероятности отказа и его соответствующие параметры. В нашем примере (см. рис. 29) мы указываем, что наступление отказа оператора описывается распределением Бернулли с вероятностью отказа 0,1.

Во время имитации генерирование событий наступления отказа будет происходить по методу Монте-Карло. Основной смысл предлагаемого нами подхода к генерированию отказов заключается в следующем:

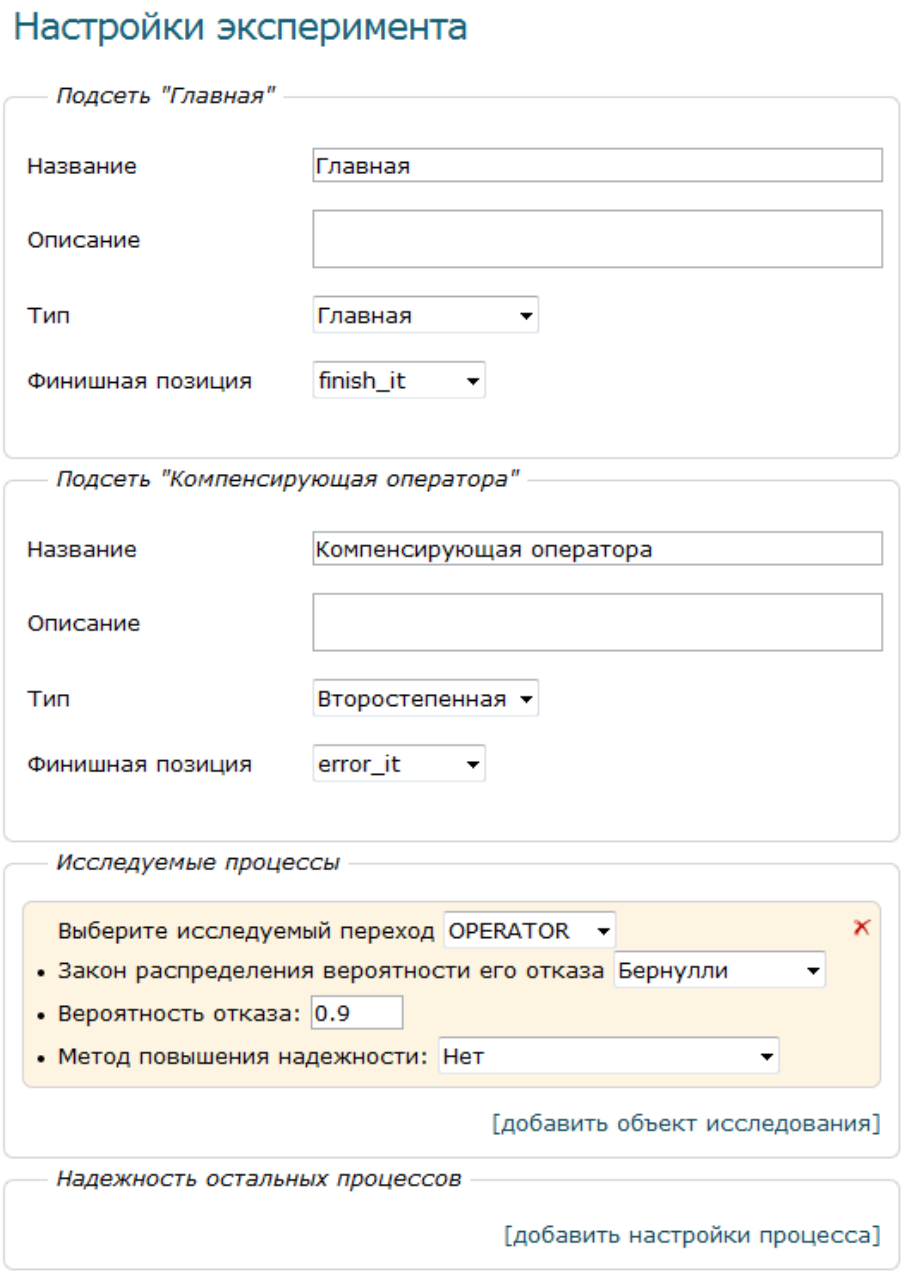


Рисунок 29 – Настройки эксперимента

– Для распределения Бернулли пусть задано, что факт наступления отказа при обращении к операции описывается этим распределением с параметром k, означающим вероятность наступления отказа, а факт отказа

отображается величиной x, которая может принимать значения 1, если отказ наступил, и 0, в противном случае. Тогда: $x = \begin{cases} 1, \text{если } bern(k) = 1 \\ 0, \text{если } bern(k) = 0 \end{cases}$, где $bern(k)$ – функция, генерирующая случайную величину, распределенную по закону Бернулли. Это распределение может использоваться, когда заранее известна вероятность отказа при вызове к операции.

- Биномиальное распределение может использоваться, чтобы показать сложное устройство, состоящее из n элементов, где отказ любого из этих элементов приводит к отказу всей операции. Вероятность отказа элементов постоянна и равна p. Тогда факт отказа x принимает значения: $x = \begin{cases} 1, \text{если } bin(n,p) \geq 1 \\ 0, \text{если } bin(n,p) = 0 \end{cases}$, где $bin(n,p)$ – функция, генерирующая случайную величину с параметрами n, p.
- Несмотря на то, что ПКБТ и в целом предложенный аппарат моделирования бизнес-транзакций не учитывает фактор времени, в ПКБТ заложена возможность генерирования отказов, которые подчиняются закону Пуассона. Для этого вводится следующее условие: пусть функция $pois(k)$ генерирует случайную величину, которая отображает количество отказов операции за единицу времени. Тогда за единицу времени принимается условное среднее время нахождения операции в работающем состоянии при обслуживании одной порции входных данных. Параметр k указывает интенсивность потока отказов (среднее количество отказов за единицу времени). Тогда факт отказа x принимает значения: $x = \begin{cases} 1, \text{если } pois(k) \geq 1 \\ 0, \text{если } pois(k) = 0 \end{cases}$.
- Если операция во время выполнения подвержена отказам, количество которых равномерно распределено на отрезке $[a, b]$, где $a, b \in \mathbb{N}$, то для генерирования отказа может использоваться равномерное распределение. Тогда факт отказа x принимает значения: $x = \begin{cases} 1, \text{если } unif(a,b) \geq 1 \\ 0, \text{если } unif(a,b) < 1 \end{cases}$, где $unif(a,b)$ – функция, генерирующая случайную величину равномерно распределенную между a, b.
- Генерирование событий отказа по другим законам распределения происходит аналогичным способом, где отказ засчитывается, если абсолютное значение случайной величины становится больше или равным 1. Нахож-

дение закона распределения и расчет необходимых параметров предполагается исследователю проводить самостоятельно.

Отметим, что в нашем примере пока компенсирующая сеть остается не задействованной (нет метода повышения надежности для операции), несмотря на то, что в схеме она определена.

Проведение имитационного эксперимента

После установки необходимых параметров ПКБТ готовит среду в выполнению имитационного моделирования. После подготовки пользователю показывается окно с уведомлением о готовности (см. рис. 30).

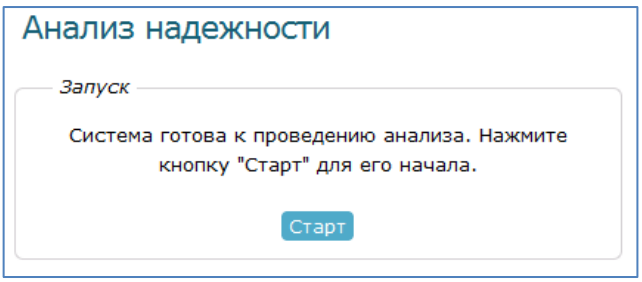


Рисунок 30 – Уведомление о готовности

После нажатия кнопки «Старт» начинается процесс имитации. Он проходит в несколько итераций, на каждой их которых происходят замеры надежности. Итерация включает полный цикл всех запусков, определенных в схеме (в нашем примере их 50), в каждом из запусков выполняется не более определенного количества переходов (в нашем примере их тоже 50). В начале каждой итерации ПКБТ изменяет загруженную схему процесса в соответствии с настройками эксперимента и ПКБТ: это может быть, например, ступенчатое изменение надежности определенного перехода для исследования степени чувствительности надежности бизнес-транзакции от его выполнения. Измененная схема бизнес-транзакции запускается на имитацию, после чего происходит анализ статистических данных, полученных в ходе итерации (см. рис. 31).

Расчет количества имитационных прогонов проходит с помощью пробного этапа моделирования и основан на следующих положениях. Во-первых, мы показали ранее, что основным количественным показателем, использующимся для расчета различных сторон надежности, является количество отказов. Зная количество отказов и общее количество запусков то можно оценить, например, вероятность отказа.

Во-вторых, за оценку количества отказов будет приниматься его среднее значение по некоторому количеству реализаций эксперимента.

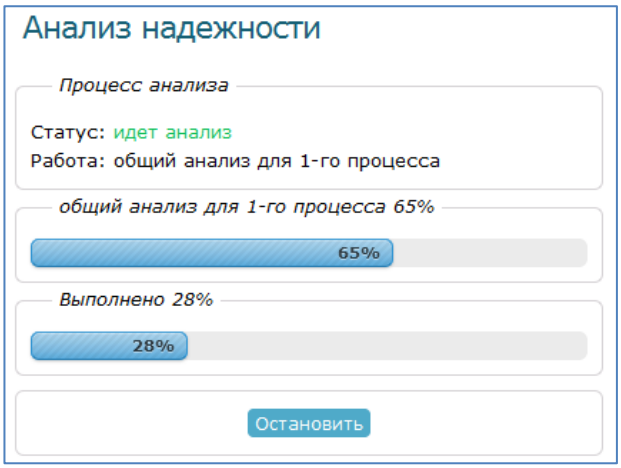


Рисунок 31 – Выполнение имитаций

В-третьих, так как получаемая оценка в силу различных факторов будет отличаться от количества отказов, введем понятие величины ε, которую назовем точностью оценки, такой что $|a - \bar{a}| < \varepsilon$, где a – количество отказов, а $\bar{a}$ – его оценка. Вероятность того, что неравенство исполняется $P(|a - \bar{a}| < \varepsilon) = \alpha$, будем называть достоверностью.

Тогда воспользуемся формулой из [153, 157, 239], где количество имитационных прогонов N представляется как:

$$N = \frac{t_{a/2}^2 * \sigma^2}{\varepsilon^2},$$

где σ^2 – оценка дисперсии, $t_{a/2}$ значение функции, обратной функции стандартного нормального распределения. Так как в ПКБТ достоверность α принимается за 0,95, то значение $t_{a/2}$ – 1,96 [240], а допустимая ошибка оценки ε принимается за 1% от максимально возможного количества отказов. Оценка σ^2 вычисляется после предварительного этапа моделирования, который включает 100 имитационных прогонов. После вычисления N запускается основной этап имитационного эксперимента.

Длительность самого процесса имитации зависит от множества факторов: вычислительной мощности операционной среды, сложности схемы процесса, типа анализа и количества анализируемых процессов. После окончания анализа ПКБТ проведет анализ суммарной информации и автоматически покажет пользователю окно с результатами анализа.

Просмотр результатов моделирования

Окно результатов анализа состоит из нескольких областей. В первую очередь (см. рис. 32) ПКБТ отображает свои выводы о результатах анализа и стати-

стику работы отдельного процесса. Выводы строятся с помощью внутренней базы фактов, собранных автором программы в результате других экспериментов, а также анализа экспертного опыта. Отдельное заключение отображается по каждому анализируемому процессу.

Анализ процесса OPERATOR показал, что восстанавливаемость всей транзакции не зависит от отказов этого процесса. Поэтому, к сожалению, восстанавливаемость транзакции невозможно улучшить, воздействуя на этот процесс. Атомарность уменьшается вместе со снижением стабильности процесса. И атомарность транзакции можно улучшить, повышая надежность работы этого процесса. Отказоустойчивость, стабильность ведут себя нелинейно при снижении стабильности процесса, и на это следует обратить внимание.

Анализ процесса "OPERATOR"

Таблица показателей надежности

	Наиболее вероятно	Минимум	Максимум	В среднем
Вероятность избежания аварий	0.90	0.76	1.00	0.90
Интенсивность отказов	0.11	0.02	0.32	0.11
Вероятность безотказной работы	0.90	0.76	1.00	0.90
Частота отказов	0.10	0.00	0.24	0.10
Средняя наработка на отказ	9.00	3.17	49.00	11.98

Рисунок 32 – Заключение о результатах анализа процесса

Вторая область окна результатов описывает выполнение транзакции в целом (см. рис. 33). Как и для отдельного процесса, здесь выводятся основные показатели надежности, данные нами в таблице 4. Каждый показатель характеризуется 4 числами: минимумом, максимумом, средним и модой. Кроме того, отображается гистограмма[1] частот отказов бизнес-транзакции, чтобы исследователь мог самостоятельно оценить распределение отказов и сделать соответствующие выводы.

Вероятность безотказной работы и частота отказов являются обоюдно-обратными величинами. Под частотой отказов здесь понимается отношение количества отказов к количеству экспериментов, что для биномиального распределения является несмещенной и состоятельной оценкой [241]. Исходя из теоремы Бернулли можно считать эту частоту вероятностью.

Следуя свойствам биномиального распределения примем во внимание, что с ростом количества испытаний в силу центральной предельной теоремы распределение отказов стремится к нормальному, а при фиксированной вероятности события и большом количестве испытаний – к распределению Пуассона.

Поэтому в ПКБТ приводятся результаты проверки гипотез (по критерию Пирсона и Колмогорова) на соответствие распределения отказов наиболее популярным в теории надежности законам распределения: Пуассона и нормального. Результаты исследований сложных бизнес-транзакций с множеством нена-

[1] Под гистограммой мы понимаем графический способ представления табличных данных, а не геометрическое отображение функции плотности вероятности

дежных элементов показали, что распределение их отказов стремится к нормальному закону, оправдывая положения центральных предельных теорем.

Анализ всей транзакции

Таблица показателей надежности

	Наиболее вероятно	Минимум	Максимум	В среднем
Атомарность (в операциях)	0.9000	0.7600	1.0000	0.9005
Вероятность устранения отказа	0.0000	0.00	0.0000	0.00
Разрешимость отказов	0.0000	0.0000	0.0000	0.0000
Вероятность безотказной работы	0.9000	0.7600	1.0000	0.8989
Вероятность безаварийной работы	0.0000			
Частота аварий	0.1000	0.0000	0.2400	0.1011
Средняя наработка на отказ	9.0000	3.1667	49.0000	11.2707
Интенсивность отказов	0.1111	0.0204	0.3158	0.1153

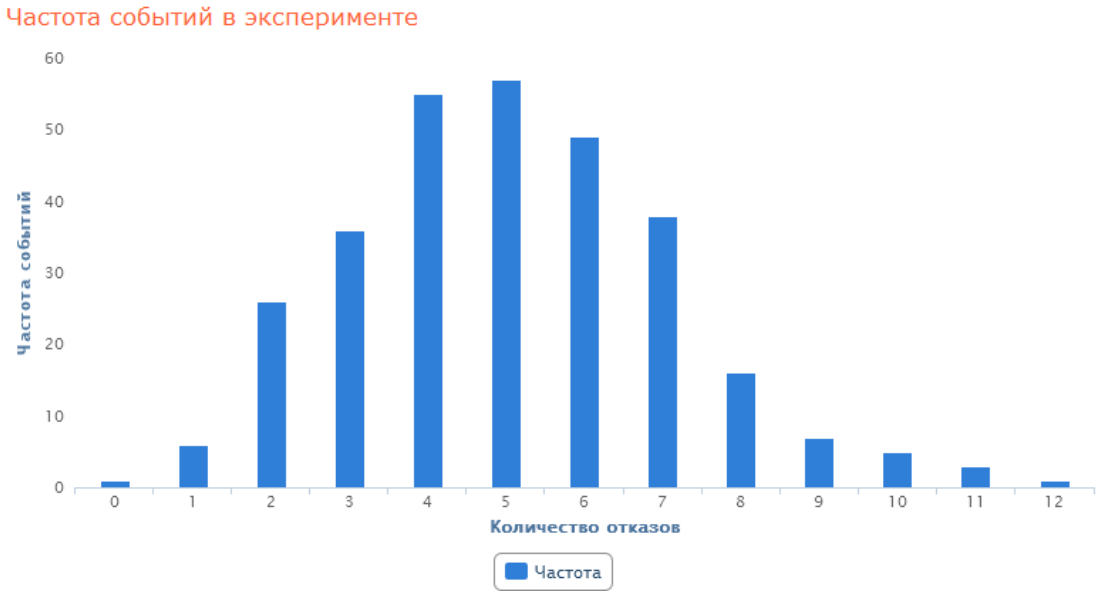


Рисунок 33 – Результаты анализа бизнес-транзакции целиком

Поэтому ПКБТ также проводит расширенный анализ гипотез на нормальное распределение с помощью критерия Колмогорова, асимметрии, эксцесса и μ-критерия (см. рис. 34). В нашем примере, несмотря на некоторую схожесть гистограммы частот отказов, гипотеза о нормальном распределении была отклонена (при уровне значимости 0,95).

Следующий блок включает отчеты анализа чувствительности. На чувствительность к отказам процессов бизнес-транзакции проверяются ключевые показатели ее надежности: разрешимость отказов, вероятность избежания аварий, атомарность, вероятность безотказной работы. Как видно из таблицы 4, эти показатели входят в 4 группы коэффициентов из предложенных 4. Частный показатель восстанавливаемости бизнес-транзакций пока в ПКБТ не рассчитывается. Это связано с тем, что для расчета показателей необходимо учитывать надежность восстанавливающих или компенсирующих сетей бизнес-транзакции, которые в данной версии ПКБТ предполагаются абсолютно надежными.

Проверка гипотез о виде распределения

Таблица результатов проверки по критерию Пирсона

Вид распределения	Гипотеза принята?
Нормальное распределение	Нет
распределение Пуассона	Нет

Проверка гипотезы о нормальном распределении

Таблица результатов проверки на нормальность

Тип теста	Гипотеза принята?
критерий Колмогорова	Нет
критерий асимметрии	Нет
μ-критерий	Нет
критерий эксцесса	Нет

Рисунок 34 – Результаты проверки гипотез на вид распределения

Кроме того, для повышения надежности восстанавливающих сетей могут использоваться очередные подсети и т.д., что приводит к необходимости итеративного наблюдения за надежностью. Так, 2.5 мы дали определение восстанавливаемой транзакции *k*-го порядка, если она состоит из восстанавливаемых операции вплоть до *k* -го порядка, но программная реализация оценки такой надежности пока не разработана.

Для анализа чувствительности бизнес-транзакции ПКБТ проводит испытания с каждым фактором на заданном в условиях уровне надежности («как есть», см. рис. 35), чередуя уровни надежности (по умолчанию – 3 уровня, см. рис. 36) и с 50% повышением надежности от заданного в условиях уровня (здесь отрабатывается сценарий возможного резервирования элемента).

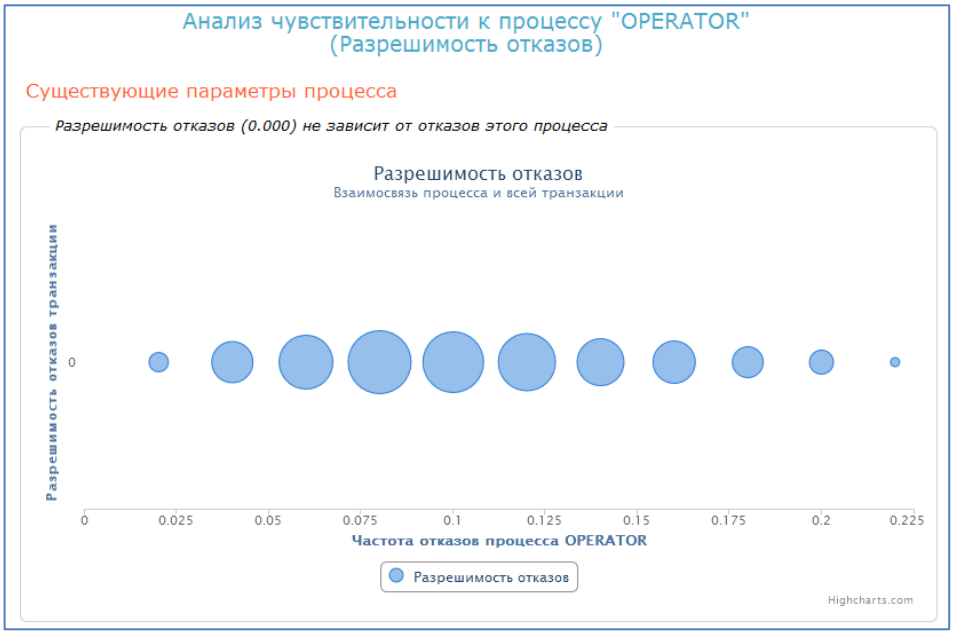


Рисунок 35 – Разрешимость отказов бизнес-транзакции при заданном уровне отказов процесса

Основным показателем надежности элемента, чувствительность к которому определяется, является частота отказов. Целесообразность расчета именного этого показателя была описана ранее.

На 35 и 36 показано, что вероятность избежания аварий и разрешимость отказов в нашем примере не зависит от частоты отказов процесса. Иными словами: факт наступления отказа процесса OPERATOR не влияет на возможность бизнес-транзакции исправлять отказы, а также любой отказ процесса OPERATOR – это авария.

Как видно с рис. 37 чем выше вероятность отказа, тем ниже атомарность всей транзакции, что легко доказывается через анализ ее схемы (см. рис. 27): видно, что процесс OPERATOR находится на «критическом пути» выполнения, и успешность его завершения прямо влияет на выполнение всей бизнес-транзакции.

Рисунок 36 – Вероятность избежания аварии при 3-х уровнях отказов процесса

Аналогичные выводы можно сделать и по вероятности безотказной работы бизнес-транзакции. Ясно, что чем чаще происходят отказы процесса OPERATOR, тем выше вероятность отказа всей транзакции (см. рис. 38). Отметим также, что при одинаковых значениях частоты отказа вероятность безотказной работы незначительно разнится, например, одинаковая вероятность может быть и при значении отказов 0,025 и 0,04.

В экспертном выводе ПКБТ об этом было сказано (см. рис. 32): «Отказоустойчивость, стабильность ведут себя нелинейно при снижении стабильности процесса, и на это следует обратить внимание».

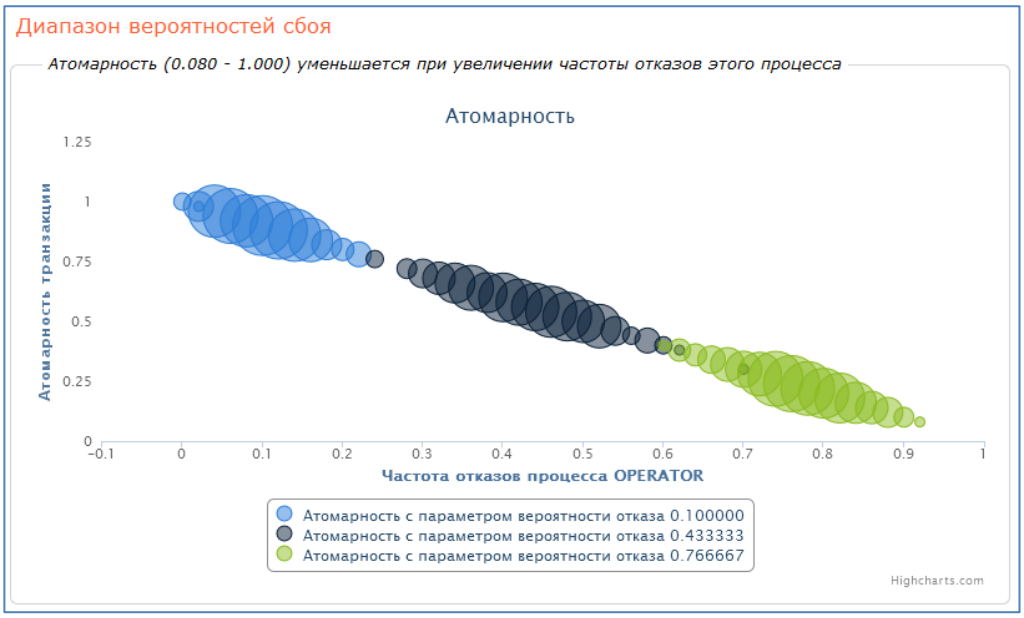


Рисунок 37 – Чувствительность атомарности при 3-х уровнях отказов процесса

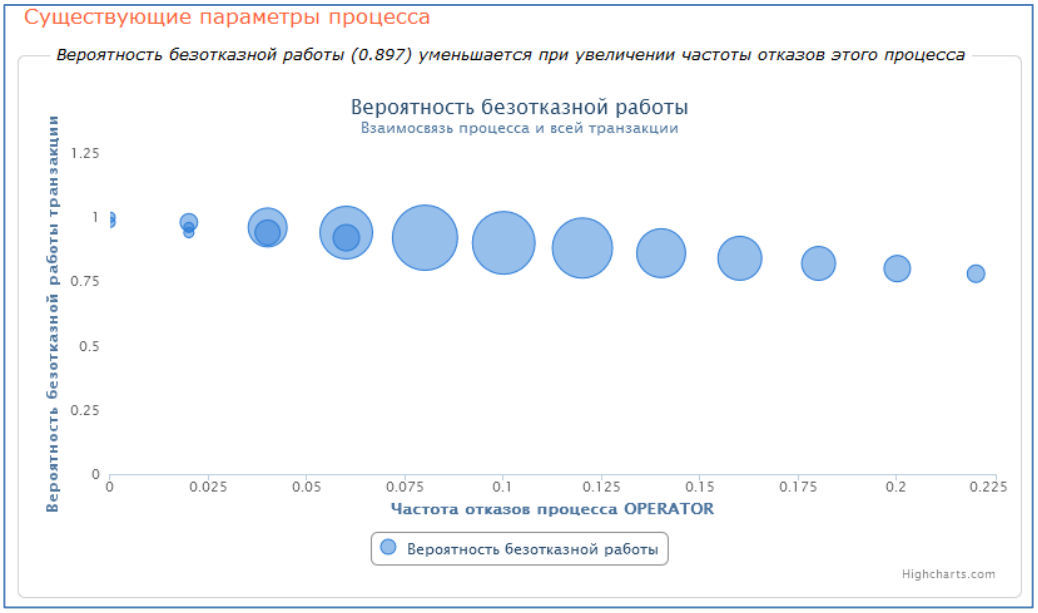


Рисунок 38 – Вероятность безотказной работы

Таким образом, представленные результаты работы ПКБТ позволяют оценить надежность бизнес-транзакции в показателях, предложенных нами ранее.

Кроме графического отображения возможной корреляции между отказами операций и показателей надежности всей транзакции, ПКБТ проводит статистический анализ этой взаимосвязи. Этот анализ проводится таким образом:

Если определяется взаимосвязь двух несвязанных выборок (например, уровней отказов операции и отказов транзакции), то:

- Производится проверка гипотезы о принадлежности выборок к нормальному распределению (для этого используются описанные ранее методы). В случае подтверждения гипотезы, проводится классический корреляци-

онный анализ. Рассчитывается коэффициент корреляции Пирсона и осуществляется проверка гипотезы о его значимости. В случае не подтверждения этой гипотезы исследуется показатель корреляционного отношения и оценивается его значимость. Таким образом, ПКБТ делает вывод о наличие линейной связи между выборками или какой-либо связи вообще.

- Если гипотеза о нормальном распределении отклоняется, то используются методы непараметрического корреляционного анализа. Рассчитывается ранговый коэффициент корреляции Спирмена и проверяется гипотеза о его значимости с помощью аппроксимации Имана-Коновера [238].

Для анализа нескольких (больше двух) выборок, например, при 3-х уровнях отказа процесса используются методы проверки различий между совокупностями данных и влияния разных уровней некоторого фактора на эти данные.

- В случае принадлежности выборок нормальному распределению используется однофакторный дисперсионный анализ, с помощью которого проверяется гипотеза о равенстве математических ожиданий отказов транзакции на разных уровнях надежности операции. Для этого проводится проверка однородности дисперсий с помощью критерия Бартлетта, а затем используется критерий Фишера для оценки изменчивости дисперсий. Неоднородность дисперсий или отклонение гипотезы о равенстве выборочных средних является показателем определенного влияния операции на транзакцию. [242]
- В случае сложности с однозначным подтверждением нормального закона распределения используются непараметрические методы, проверяющие значимость изменения математического ожидания нескольких совокупностей. Если проводится сравнение нескольких несвязанных выборок (например, для сравнения изменений, внесенных исследователем в схему бизнес-транзакции для улучшения ее показателей), то используется непараметрический критерий сдвига Краскала-Уоллиса. С помощью него проверяется гипотеза о наличии сдвига, а для объема выборки > 5 применяется более точная [238] аппроксимация Имана-Давенпорта. Если выборки связанные, то используется критерий Фридмена-Кендалла-Бэбингтона Смита с аппроксимацией Имана-Давенпорта.

Отметим, что ПКБТ на данном этапе своего развития использует метод исследования «один фактор в один момент времени», в котором все операции поочередно изучаются на влияние на надежность транзакции при фиксации

надежности остальных операций. Недостатки этого метода известны, однако проведение симметричного факторного эксперимента для исследования бизнес-транзакции считается нами нецелесообразным ввиду следующих причин:

- Положения дисперсионного и регрессионного анализа требуют независимости операций, влияющих на общую надежность транзакции. Очевидно, что часто бизнес-транзакция включает последовательно соединенные операции, что обязывает исследовать эффекты взаимодействия между факторами.
- По тем же причинам пренебрежение взаимодействием высоких порядков нецелесообразно, что не позволяет построить дробные планы эксперимента.
- Попарное изучение факторов могло бы позволить исключить взаимозависимые, но в случае с последовательным подключением операций такое исключение приведет к невозможности изучения важных операций, так как последующая операция, естественно, зависит от предыдущей в структуре, но исключать ни первую, не вторую нельзя.
- Изучение зависимостей между факторами часто требует предпосылок в виде (совместного) нормального или другого известного распределения. В общем случае, распределение частоты отказов операции и транзакции может быть различным.

Попытка применения методов факторного эксперимента, регрессионного или дисперсионного анализа направлены на построение функции зависимости надежности от операций. Однако в теории надежности такая функция называется структурной функцией надежности, и она может быть легко построена по схеме системы. Очевидную сложность здесь представляет нетривиальные зависимости между операциями, выраженные условиями, циклами и индивидуальностью операций, что может не позволить построить точную структурную функцию.

Таким образом, сложность бизнес-транзакции в общем случае не позволяет использовать формальные аналитические методы для исследования зависимости ее надежности от операций.

Эксперимент с внедрением восстанавливающей сети

Воспользуемся возможностью продемонстрировать преимущества сервис-ориентированной среды и подключить для исправления отказов OPERATOR восстанавливающую сеть. Вход в восстанавливающую сеть в схеме процесса

начинается с позиции «reset». В прошлый раз эта сеть не была задействована и при имитации не работала. Для использования этой сети добавим ее как средство повышения надежности процесса OPERATOR (см. рис. 39). Важным моментом является то, что исследователь должен самостоятельно указать, какие данные необходимо передать от процесса к позиции входа восстанавливающей сети.

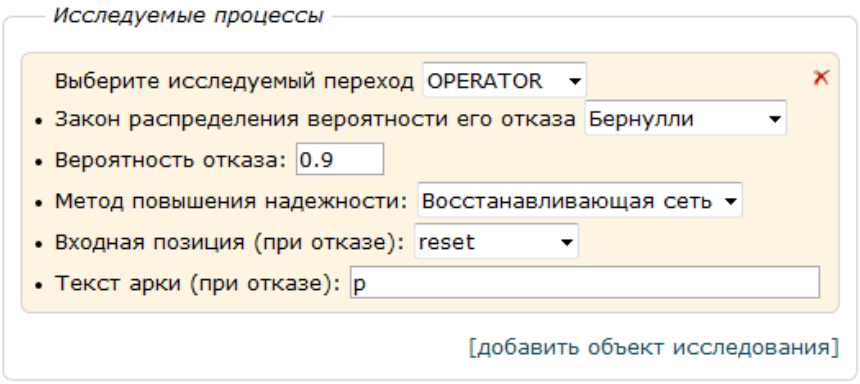


Рисунок 39 – Настройка эксперимента с использованием восстанавливающей сети

Результаты эксперимента с новыми параметрами позволяют говорить о практически абсолютной надежности бизнес-транзакции, т.к. единственный не абсолютно надежный элемент OPERATOR в случае отказа дублируется абсолютно надежной восстанавливающей сетью (см. рис. 40):

Анализ процесса "OPERATOR"

Таблица показателей надежности

	Наиболее вероятно	Минимум	Максимум	В среднем
Вероятность избежания аварий	1.00	1.00	1.00	1.00
Интенсивность отказов	0.11	0.02	0.32	0.11
Вероятность безотказной работы	0.90	0.76	1.00	0.90
Частота отказов	0.10	0.00	0.24	0.10
Средняя наработка на отказ	9.00	3.17	49.00	12.07

Анализ всей транзакции

Таблица показателей надежности

	Наиболее вероятно	Минимум	Максимум	В среднем
Разрешимость отказов	1.00	1.00	1.00	1.00
Вероятность избежания аварий	1.00	0.50	1.00	1.00
Атомарность	1.00	1.00	1.00	1.00
Интенсивность отказов	0.00	0.02	0.02	0.02
Вероятность безотказной работы	1.00	0.98	1.00	1.00
Частота отказов	0.00	0.00	0.02	0.00

Рисунок 40 – Результаты эксперимента при использовании восстанавливающей сети

Добавим условие, что восстанавливающая сеть надежна только с вероятностью 0,95, для этого в группе «Надежность остальных процессов» добавим условие надежности перехода DOUBLER, который, как известно из схемы при-

мер, является единственным переходом восстанавливающей сети (см. рис. 41). Пусть переход DOUBLER отличается низкой надежностью с вероятностью отказа 0,7.

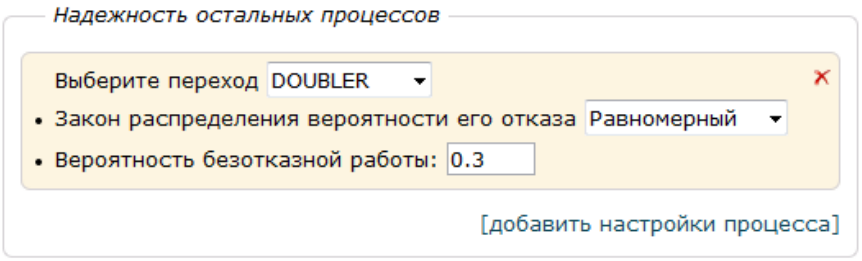


Рисунок 41 – Определение уровня надежности восстанавливающей сети

Заметим, что характеристики надежности процесса OPERATOR практически не изменились, однако надежность транзакции целиком снизилась: вероятность избежания аварий снизилась с 1,0 до 0,32 (в среднем), интенсивность отказов повысилась, а вероятность безотказной работы снизилась (см. рис. 42).

Анализ процесса "OPERATOR"

Таблица показателей надежности

	Наиболее вероятно	Минимум	Максимум	В среднем
Вероятность избежания аварий	1.00	1.00	1.00	1.00
Интенсивность отказов	0.11	0.02	0.28	0.12
Вероятность безотказной работы	0.90	0.78	1.00	0.90
Частота отказов	0.10	0.00	0.22	0.10
Средняя наработка на отказ	9.00	3.55	49.00	11.41

Анализ всей транзакции

Таблица показателей надежности

	Наиболее вероятно	Минимум	Максимум	В среднем
Разрешимость отказов	1.00	1.00	1.00	1.00
Вероятность избежания аварий	0.00	-0.20	1.00	0.32
Атомарность	1.00	1.00	1.00	1.00
Интенсивность отказов	0.09	0.02	0.19	0.08
Вероятность безотказной работы	0.92	0.84	1.00	0.93
Частота отказов	0.08	0.00	0.16	0.07
Средняя наработка на отказ	11.50	5.25	49.00	17.67

Рисунок 42 – Результаты эксперимента при использовании ненадежной восстанавливающей сети

Корреляционное поле демонстрирует больший разброс значений (см. рис. 43): вероятность избежания аварий теперь колеблется от 0 до 1, сосредотачиваясь вокруг значения 0,3 – это легко объяснимо тем, что вероятность отказа восстанавливающей транзакции также 0,3.

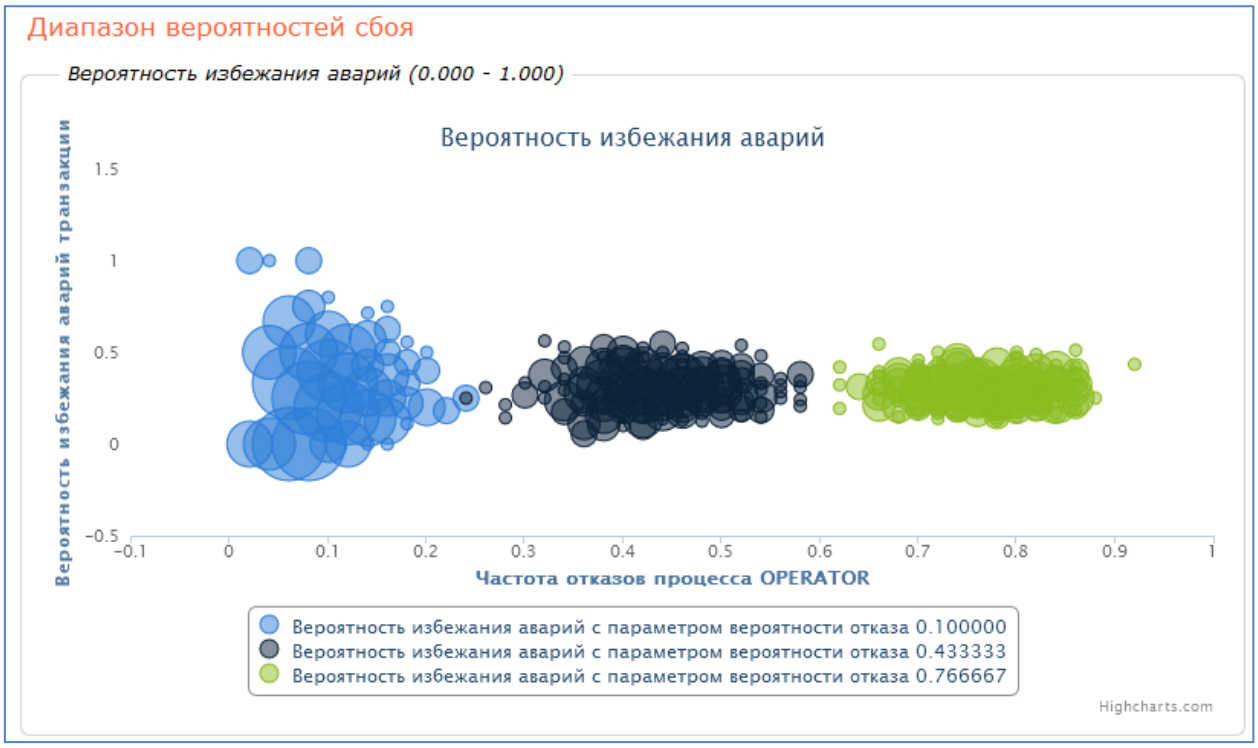


Рисунок 43 – Зависимость вероятности избежания аварий от частоты отказов

Работа с компенсирующими сетями реализуется аналогично, только исследователю необходимо указать еще и выходную позицию, по которой будет отслеживаться успешность компенсируемых действий (см. рис. 44).

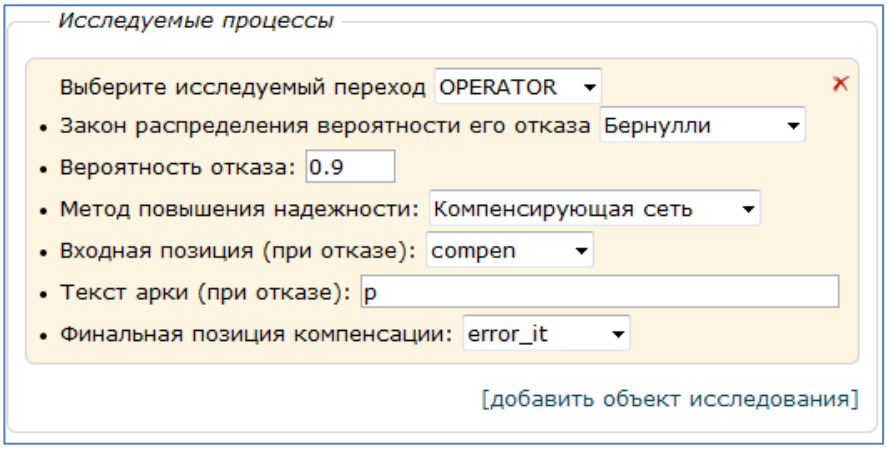


Рисунок 44 – Настройка компенсирующей сети

Анализ устойчивости бизнес-транзакции

Продемонстрируем возможности анализа устойчивости бизнес-транзакции с помощью цепей Маркова.

В отличие от других типов исследований, анализ устойчивости предполагает контроль за позициями бизнес-транзакции. Так как количество возможных маркировок бизнес-транзакции экспоненциально возрастает с количеством позиций, то для проведения этого анализа часто бывает необходимо искусственно

ограничивать количество отслеживаемых позиций. Иначе при большом количестве маркировок алгоритм нахождения фундаментальной матрицы, и а именно обратной матрицы от разницы матриц Q и I становится настолько сложнореализуемым, что его требования превышают вычислительные возможности современных персональных компьютеров.

Поэтому перед этим анализом необходимо:

- Убедиться, что бизнес-транзакция не содержит операций, генерирующих случайные данные. Иначе между имитациями не будет одинаковых маркировок.
- Убедиться, что запускается один экземпляр бизнес-транзакции в каждую имитацию. Иначе случайное распределение данных нескольких бизнес-транзакций приведет к неконтролируемо большому количеству разных маркировок.
- Исключить из мониторинга те позиции, которые не влияют на выполнение бизнес-транзакции.

Выполнение данных условий требует модификации схемы бизнес-транзакции. В рассматриваемом примере необходимо только модифицировать позицию start_it и переход IN. Удалим возможность случайно генерировать входной номер (см. рис. 45).

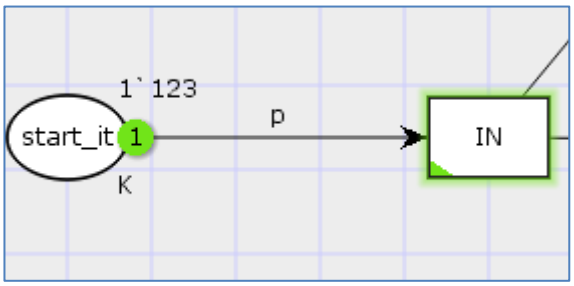


Рисунок 45 – Модификация входной позиции

Так как позиций в бизнес-транзакции сравнительно немного (ограничение ПКБТ – не более 200 маркировок), то не будем ничего исключать из анализа.

Результат исследования устойчивости в ПКБТ состоит из 3 частей (см. рис. 46):

- Собственно результаты исследования устойчивости в виде простой текстовой строки. Если результаты неудовлетворительные, то описываются позиции, которые составляют возвратные маркировки.
- Анализ достижимости финальной маркировки, и, что немаловажно, подтверждение того, что бизнес-транзакция останавливает свою работу на ней.

– Граф переходов состояний системы. Зеленым отображается начальная маркировка, красным – группа финальных, синим – обычные маркировки, а серым – состояние, обозначающее завершение бизнес-транзакции. При наведении указателя на узел графа ПКБТ описывает структуру маркировки.

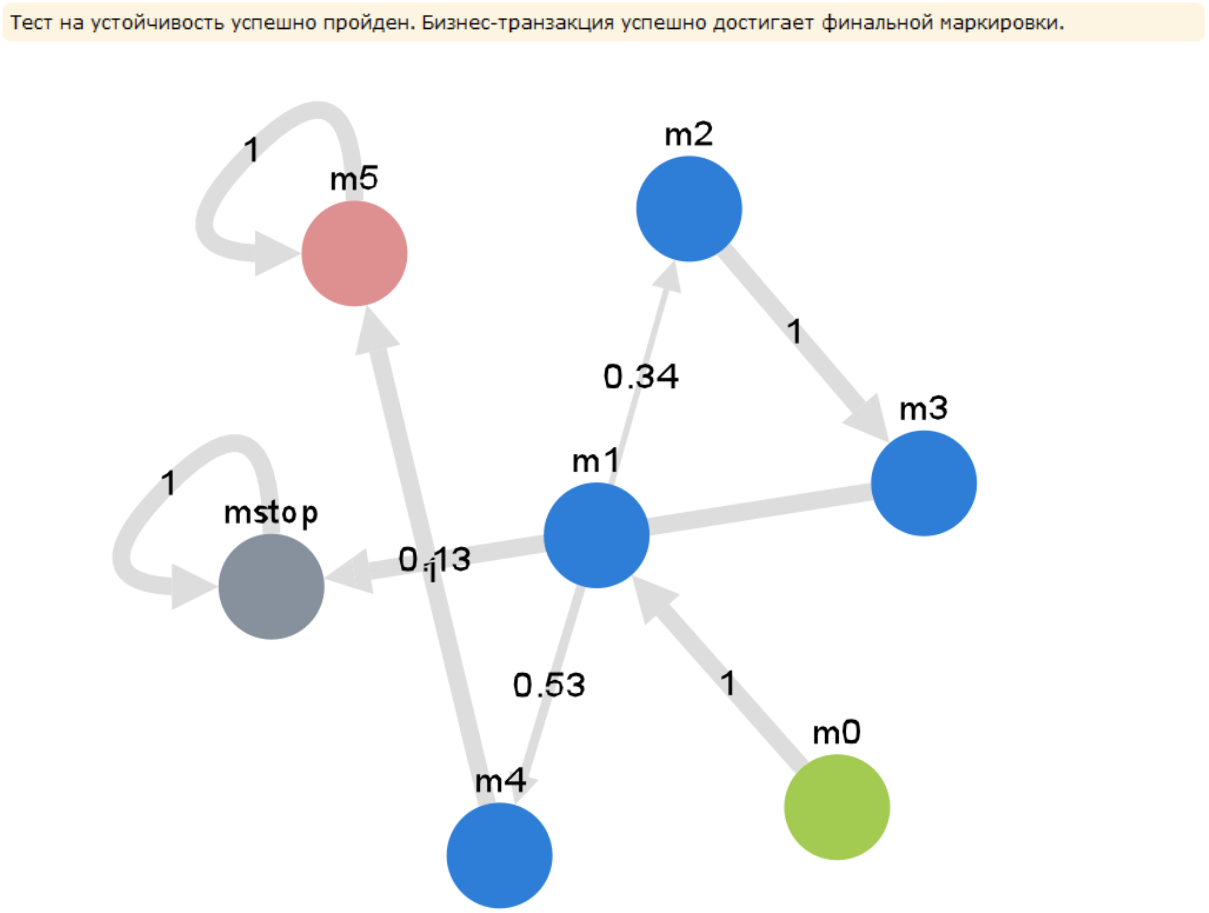


Рисунок 46 – Результаты анализа устойчивости с помощью цепей Маркова

Результат анализа примера показывает, что бизнес-транзакция устойчива.

Таким образом, применение ПКБТ для анализа надежности бизнес-транзакции не является более сложной процедурой, чем, например, построение схемы процесса в нотации BPMN и использование этой схемы в BPM-системах. Это позволяет использовать ПКБТ широким кругом лиц, не обладающими специфическими знаниями и навыками в области имитационного моделирования, эконометрики, теории надежности или программировании.

Приложение Б. Сравнение стандартных методов расчета надежности и ПКБТ

Проведем сравнение результатов оценки надежности, полученных с помощью популярных методов расчета и ПКБТ. И продемонстрируем, что аналитические методы расчета надежности усложняются при росте сложности исследуемой системы.

Описание анализируемой бизнес-транзакции

Предположим, что необходимо рассчитать надежность бизнес-транзакции со следующей простой структурной схемой:

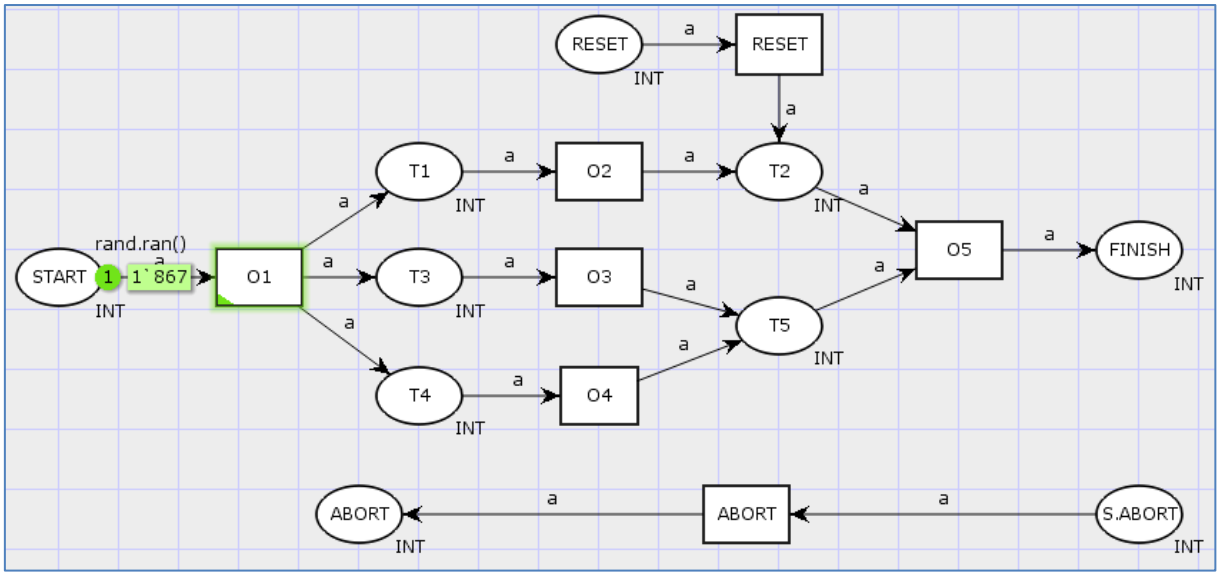


Рисунок 47 – Структурная схема бизнес-транзакции

Бизнес-транзакция представляет собой упрощенную модель бизнес-процесса, где происходит передача некоторой переменной (например, маркера управления) между операциями. В таком виде бизнес-транзакция может быть записана даже на языке простых сетей Петри, быть представима диаграммами UML, IDEF0 или BPMN. Структурные методы расчета надежности исследуют именно такой тип систем, где связи между элементами имеют явную направленность, и нет сложной логики передачи данных или управления.

Подсеть «ABORT – S.ABORT» является компенсирующей операцию O5, а позиция «RESET» – начинает восстанавливающую подсеть для операции O2. Проведем расчеты надежности с помощью описанных методов и сравним результаты с результатами работы ПКБТ. Будем учитывать, что структурные методы расчета надежности дискретных систем опираются на вероятность отказа системы как основной показатель надежности, и ПКБТ способен также рассчи-

тывать этот показатель, так как он входит в группу рекомендуемых нами для анализа надежности бизнес-транзакции. Ясно, что компенсирующая подсеть не влияет на показатель вероятности отказа (т.е. является несущественным элементом), поэтому нами она не будет рассматриваться при расчетах надежности.

Зададим следующие вероятности отказа элементов бизнес-транзакции (см. табл. 22).

Таблица 22 – Вероятности отказа элементов бизнес-транзакции

Название операции	Вероятность отказа q	Вероятность работоспособности $p = 1 - q$
O1	0,05	0,95
O2	0,1	0,9
O3	0,08	0,92
O4	0,07	0,93
O5	0,15	0,85
RESET	0,05	0,95

Понятие монотонной структурной функции системы

Методы расчета надежности дискретных систем опираются на понятие монотонных структурных функций. Введем несколько определений. Условия работоспособности системы – это условия, при которых она может выполнить стоящую перед ней задачу. Под структурной схемой системы будем понимать графическое отображение структурных элементов системы и связей между ними, предназначенное для отображения условия работоспособности системы [99]. Структурный элемент – это условный элемент, обладающий теми же характеристиками надежности, что и реальный элемент системы. В некоторых случаях структурный элемент может отображать совокупность последовательно-соединенных элементов.

Пусть каждый структурный элемент системы может находиться в одном из двух состояний: полной работоспособности и отказа. Введем x_i – переменную, соответствующую i-му структурному элементу, которая может принимать два значения:

$$x_i = \begin{cases} 1, \text{если } i - \text{й элемент работоспособен} \\ 0, \text{если } i - \text{й элемент отказал} \end{cases}$$

Ясно, что вероятность работоспособного состояния такого элемента может определяться как математическое ожидание $p_i = M[x_i]$. Пусть n-мерный вектор $X = (x_1, x_2, \dots, x_n)$ характеризует состояние системы из n элементов. Зададим функцию от этого вектора

$$\varphi(X) = \varphi(x_1, x_2, \dots, x_n),$$

которую называют функцией работоспособности, условиями работоспособности или структурной функцией системы. Число n называется порядком структурной функции [213]. Эта функция также может принимать значения:

$$\varphi(X) = \begin{cases} 1, \text{если система работоспособна} \\ 0, \text{если система отказала} \end{cases}$$

Вероятность работоспособного состояния системы может определяться как математическое ожидание структурной функции: $\mathcal{P} = M[\varphi(X)]$.

Структурный элемент системы i будем называть несущественным в структуре, если φ не зависит от значения x_i, т.е. $\varphi(x_1, x_2, \dots, 0_i, \dots, x_n) = \varphi(x_1, x_2, \dots, 1_i, \dots, x_n)$. Иначе элемент будем называть существенным.

Структура системы является монотонной (или связанной, или когерентной [99]), если выполняются следующие условия [213, 95]:

- $\varphi(X^0) = 0$, где $X^0 = (0,0, \dots .,0)$
- $\varphi(X^1) = 1$, где $X^1 = (1,1, \dots .,1)$
- $\varphi(X) \geq \varphi(Y)$, где $x_i \geq y_i$ для $i = 0 \dots n$

Понятно, что каждый элемент такой системы – существенный. Таким образом, наиболее популярные методы расчета надежности основываются на предположении, что каждый элемент системы может находиться в двух состояниях, в системе нет элементов, которые не влияют на ее работоспособность, а структурная функция возрастает с ростом работоспособности элементов.

Очевидно, что для системы, структура которой состоит из последовательно подключенных элементов, где отказ любого приводит к отказу системы, функция работоспособности имеет вид:

$$\varphi(X) = \prod_{i=1}^{n} x_i = \min(x_1, x_2, \dots, x_n)$$

Для системы из параллельных элементов:

$$\varphi(X) = 1 - \prod_{i=1}^{n} (1 - x_i) = \max(x_1, x_2, \dots, x_n)$$

Вычисления с помощью структурной функции требуют использования аппарата алгебры логики, поэтому зададим ряд базовых операций. Отрицанием x будем называть функцию, обозначаемую $\bar{x}$, значение которой противоположно значению x. Например, $\bar{0} = 1$. Конъюнкцией (или логическим умножением) двух переменных x и y будем называть операцию, обозначаемую как $x \wedge y$ (или просто xy), которая возвращает 1 только в том случае, если оба аргумента равны 1, в остальных случаях – 0. Иными словами: $1 \wedge 1 = 1$, но $1 \wedge 0 = 0$. Дизъ-

юнкцией (логическим сложением) будем называть операцию, обозначаемую как $x \vee y$, которая возвращает 0 только в том случае, если оба аргумента равны 0, в остальных случаях – 1. Иными словами: $0 \vee 0 = 0$, но $1 \vee 0 = 1$. При записи формул операция конъюнкции имеет приоритет над дизъюнкцией, т.е. $z \vee x \wedge y = z \vee (x \wedge y)$, а отрицание имеет приоритет над двумя другими.

Тогда структурная функция последовательно подключенных элементов будет иметь вид: $\varphi(X) = x_1 \wedge x_2 \wedge \ldots \wedge x_n$, а параллельно подключенных: $\varphi(X) = x_1 \vee x_2 \vee \ldots \vee x_n$.

Все методы, использующие структурную функцию, предполагают последовательное построение этой функции для конкретной системы с последующей заменой переменных x на соответствующие вероятности работоспособности элементов, что приводит к получению вероятности работоспособности всей системы. Т.е. происходит переход к логической функции к вероятностной функции, которая показывает вероятность истинности структурной функции [99]. Рассмотрим ряд этих методов.

Логико-вероятностный метод

Логико-вероятностный метод расчета структурной надежности (схемно-логический метод) предполагает построение функции работоспособности с использованием алгебры логики, учитывая характер взаимодействия элементов системы. Наиболее часто этот метод применяют для отображения структурной схемы системы в функцию его работоспособности. Полученную сложную логическую функцию с помощью свойств логических операций можно привести к минимальной бесповторной форме, а затем арифметизировать, заменив операции конъюнкции, дизъюнкции и отрицания на соответствующие операции над вероятностями отказа / работоспособности операций.

Проведем расчет надежности с помощью логико-вероятностного метода. Группа операций О3 и О4 выполняются параллельно, то есть результат их выполнения будет истинным в любом случае, и структурная функция группы будет:

$$\varphi(O4, O3) = O4 \vee O3.$$

Процесс О2 зарезервирован с помощью восстанавливающей сети во главе с операцией RESET, и, на первый взгляд, соотношение между ними аналогично О3 и О4, однако восстанавливающая сеть запускается только в том случае, если О2 отказал, то есть имеет место следующая запись:

$$\varphi(O2, RESET) = O2 \vee \overline{O2} \wedge RESET.$$

Для выполнения операции O5, согласно требованиям сети Петри, необходима работоспособность и ветки O2-RESET, и ветки O3-O4, поэтому они связаны соотношением конъюнкции. Кроме того, отказ операции O1 или O2 также влечет отказ всей системы. Итоговая структурная функция системы имеет вид:

$$\varphi(BT) = O1(O4 \vee O3)(O2 \vee \overline{O2} \wedge RESET)O5$$

Перейдем к вероятностной функции, принимая за p вероятность работоспособности элемента:

$$\mathcal{P}(\varphi(BT) = 1) = p(O1) * \big(p(O4) + p(O3) - p(O4) * p(O3)\big) *$$
$$* \Big(p(O2) + \big(1 - p(O2)\big) * p(RESET)\Big) * p(O5)$$

При этом мы учитывали, что работа O3 и O4 – совместные события, а O2 и RESET – несовместные.

Численное значение показателя надежности будет: 0,79896311.

Метод с использованием цепей Маркова

Метод с использованием цепей Маркова предполагает расчет вероятностей переходов между состояниями системы при ее движении от начального к конечному состоянию. Составленная матрица переходов за 1 шаг позволяет рассчитать вероятность успешной работы системы по прошествии n шагов. Другой подход заключается в составлении матрицы переходов между состояниями системы в процессе ее непрерывной эксплуатации, когда нет выраженного начального или завершающего состояния работы. Этот подход часто используется для анализа резервированных систем с восстановлением элементов. Математическое описание цепей Маркова и этого метода расчета было дано в 2.7.

Используем метод цепей Маркова. Предположим, что необходимо рассчитать вероятность перехода бизнес-транзакции из начальной к финальной маркировки. Это достаточно легко, так как в нашем примере бизнес-транзакция оперирует только одной переменной, которая последовательно передается между операциями, т.е. начальная и финальная маркировка может быть описана нахождением переменной в позициях START и FINISH соответственно. Выделим следующие маркировки (состояния) бизнес-транзакции (см. табл. 23):

Таблица 23 – Описание состояний цепи Маркова

Маркировка	Переменная в позициях
M0	START
M1	T1, T3, T4
M2	T2, T5
M3	FINISH

Продолжение таблицы 23

Маркировка	Переменная в позициях
M4	ABORT
F	FAULT

Позиция FAULT введена нами как фиктивная позиция, которая означает отказ системы. Будем считать, что в позицию FAULT переходить метка-переменная в том случае, если операция отказывает. Позиции FAULT, ABORT, FINISH являются поглощающими.

Матрица переходных вероятностей бизнес-транзакции для шага n=1 выглядит так (см. табл. 24):

Таблица 24 – Матрица переходных вероятностей 1-го шага

	M0	M1	M2	M3	M4	F
M0	0	$p(O1)$	0	0	0	$1-p(O1)$
M1	0	0	$p((O2 \vee \overline{O2} \wedge RESET) \wedge (O4 \vee O3))$	0	0	$1-p((O2 \vee \overline{O2} \wedge RESET) \wedge (O4 \vee O3))$
M2	0	0	0	$p(O5)$	$1-p(O5)$	0
M3	0	0	0	1	0	0
M4	0	0	0	0	1	0
F	0	0	0	0	0	1

или

	M0	M1	M2	M3	M4	F
M0	0	0,95	0	0	0	0,05
M1	0	0	0,989428	0	0	0,010572
M2	0	0	0	0,85	0,15	0
M3	0	0	0	1	0	0
M4	0	0	0	0	1	0
F	0	0	0	0	0	1

Согласно свойствам матрицы переходных вероятностей, найдем вероятность перехода из состояния M0 в состояние M3 за 3 шага как третью степень матрицы переходных вероятностей (см. табл. 25):

Таблица 25 – Матрица переходных вероятностей для 3-его шага

	M0	M1	M2	M3	M4	F
M0	0	0	0	0,798963	0,140993	0,060043
M1	0	0	0	0,841014	0,148414	0,010572
M2	0	0	0	0,85	0,15	0
M3	0	0	0	1	0	0
M4	0	0	0	0	1	0
F	0	0	0	0	0	1

Как мы видим, вероятность перехода равняется 0,798963, что соответствует результатам, полученным в предыдущем методе.

Метод минимальных путей и сечений

Метод минимальных путей и сечений позволяет находиться верхнюю и нижнюю границу работоспособности системы [243]. А для простых систем метод вообще позволяет точно оценивать ее надежность.

Минимальный путь функционирования системы представляет собой такую последовательность (конъюнкцию) ее элементов, когда ни один из элементов нельзя изъять без нарушения работоспособности системы. Короче говоря, отказ любого элемента на минимальном пути приводит к отказу всей системы. Минимальное сечение системы представляет собой конъюнкцию отрицаний таких элементов, что ни один нельзя изъять, не нарушив условия отказа системы. То есть система работоспособна до тех пор, пока работоспособен хотя бы один элемент из минимального сечения. Каждая система может иметь несколько минимальных путей и минимальных сечений.

С каждым минимальным путем A_j можно связать структурную функцию $\varphi_j(A_j) = \prod_{i \in A_j} x_i$, которая принимает значение 1, если все элементы пути работоспособны, и 0, если хотя бы один элемент отказал. Очевидно, что такая функция $\varphi_j(X)$ является структурной функцией системы, элементы которой соединены последовательно. Аналогично можно представить минимальное сечение B_k: $\varphi_k(B_k) = 1 - \prod_{i \in B_k}(1 - x_i)$.

Структурная функция монотонной системы в терминах минимальных путей или сечений выглядит так:

$$\varphi(X) = 1 - \prod_j (1 - \varphi_j(A_j))$$

или

$$\varphi(X) = \prod_j \varphi_j(B_k).$$

То есть систему можно представить в виде параллельно соединенных путей или последовательно-соединенных сечений.

Воспользуемся методом сечений для нахождения нижней границы вероятности безотказной работы. Согласно определению, минимальным сечением называется набор элементов, отказ любого из которых приводит к отказу системы. Восстановление работоспособности любого из элементов сечения восста-

навливает работоспособности системы. Минимальными сечениями здесь мы признаем группы операций: $\{O1\},\{O5\},\{O2,RESET\},\{O3,O4\}$. Соответствующая структурная функция выглядит так: $\varphi(BT)=\big(1-(1-O1)\big)*\big(1-(1-O5)\big)*\big(1-(1-O2)(1-RESET)\big)*\big(1-(1-O3)(1-O4)\big)$, или, переходя к вероятности: $\mathcal{P}(\varphi(BT)=1)=0{,}95*0{,}85*0{,}995*0{,}9944=0{,}798963$. Метод позволяет получить аналогичные результаты.

Метод дерева отказов

Метод дерева отказов предполагает рассматривать функционирование системы с помощью дерева событий, которое применительно к анализу отказов называют, собственно, деревом отказов. Целью построения дерева отказов является отображение причинно-следственных связей между отдельными событиями в системе, которые могут привести к отказу всей системы. Для этого одиночные события отображают в виде узлов ориентированного графа – дерева, корнем которого является результирующее событие – полный отказ всей системы. Узлы дерева соединены дугами, которые отображают причинно-следственные связи. С помощью дерева отказов получают: систематизированное представление всех причин центрального (вершинного) события, структурированный материал для анализа причин отказов, параметры надежности системы, полученные в результате анализа узлов дерева [215]. Все вершины дерева отказов делят на: вершины отказов (первичных или вторичных), а также вершины, соответствующие логическим операциям конъюнкции и дизъюнкции. Существуют различные предложения по графическому отображению этих узлов и применения деревьев к анализу отказов [215, 214, 244, 213, 105]. Вершинное событие представляется в виде логической функции из аргументов, описывающих узлы дерева. Каждый аргумент описывает одно событие, и может, как и переменная состояния структурного элемента принимать булевое значение 0 (событие не наступило) или 1 (событие наступило). Переход к вероятностной функции отказа системы осуществляется на базе тех же принципов, что мы описывали ранее.

Продемонстрируем решение задачи методом дерева отказов. Построим дерево отказов бизнес-транзакции (см. рис. 48).

Из дерева видно, что отказ ветви O2 должен произойти вместе с отказом RESET, чтобы бизнес-транзакция отказала, но RESET не сможет отказать без отказа O2 – попросту он не будет запущен, пока O2 не откажет. Соответствую-

щая дереву функция вероятности имеет вид: $\mathcal{P}(\varphi(BT) = 1) = \big(1 - q(O2) * q(RESET)\big) * \big(1 - q(O3) * q(O4)\big) * \big(1 - q(O1)\big) * \big(1 - q(O5)\big) = 0{,}79896311.$

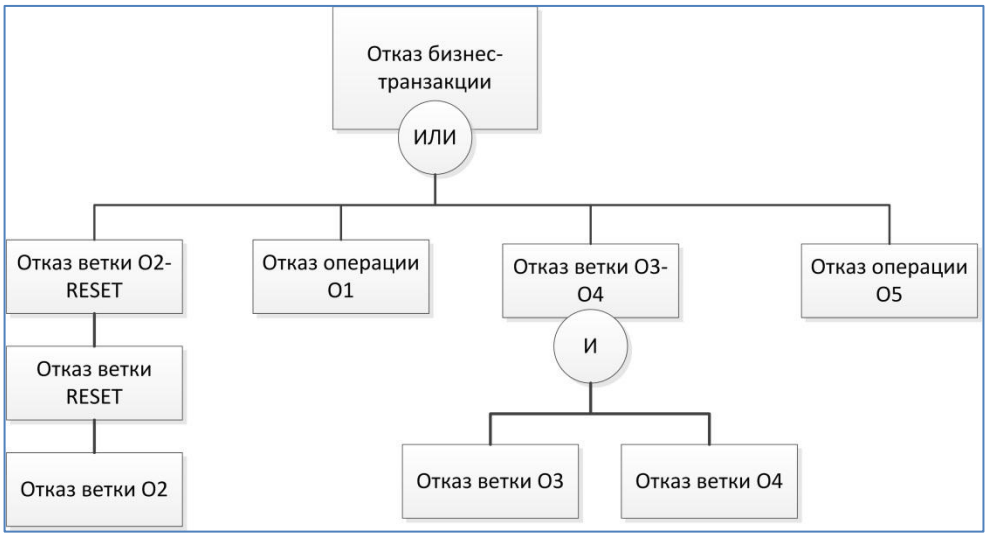


Рисунок 48 – Дерево отказов бизнес-транзакции

Расчет надежности с помощью ПКБТ

Проведем расчеты надежности бизнес-транзакции с помощью ПКБТ, согласно описанной ранее методике. Результаты расчета с помощью ПКБТ (см. рис. 49) показали вероятность безотказной работы на уровне 0,7990.

Разрешимость отказов	1.0000	0.0000	1.0000	0.5481
Вероятность безотказной работы	1.0000	0.0000	1.0000	0.7990
Вероятность безаварийной работы	1.0000	0.0000	1.0000	0.9445
Частота отказов	0.0000	0.0000	1.0000	0.2010

Интервальная оценка вероятности отказов

С вероятностью 95% вероятность отказа находится в данном интервале

Нижняя граница	Отказы	Верхняя граница
0.781013337410443	0.798600699650175	0.816188061889907

Рисунок 49 – Результаты расчета с помощью ПКБТ

Этот результат соответствует полученной по результатам исследования интервальной оценке вероятности отказа и приближается к теоретическим расчетам с точностью до 0,001. Таким образом, была точность работы ПКБТ соответствует результатам, полученным аналитическими методами.

Метод разложения относительно базового элемента

Увеличение структурной сложности системы, как мы уже говорили, ведет к нелинейному росту сложности применения описанных аналитических методов расчета надежности. Вводятся определенные допущения, количество перемен-

ных модели возрастает и снижается точность ее применения. Например, внедрим в сеть мостиковую структуру (см. рис. 50). Для расчета надежности подобной системы необходимо применить разложение по базовому элементу.

Метод разложения относительно базового элемента основан на теореме разложения функции логики по любому аргументу. Для структурной функции это предполагает истинность следующего равенства: $\varphi(x_1, x_2, \dots, x_n) = x_i\varphi(x_1, x_2, \dots, 1, \dots, x_n) \vee \overline{x_i}\varphi(x_1, x_2, \dots, 0, \dots, x_n)$. С позиции вероятностной функции метод опирается на теорему о сложении вероятностей двух несовместных событий. Этот метод позволяет свести сложные структуры, например, т.н. «мостиковые» к последовательно-параллельным логическим схемам.

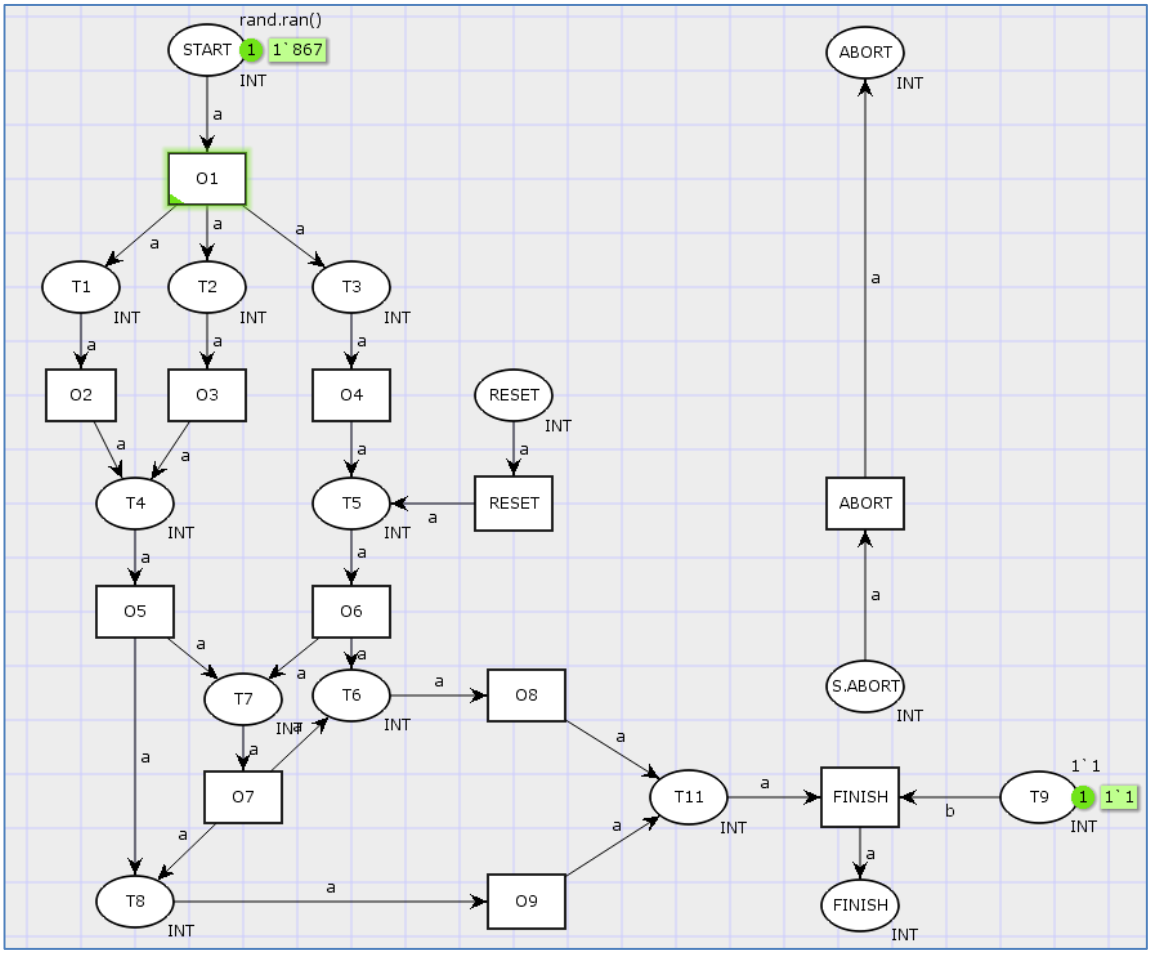


Рисунок 50 – Бизнес-транзакция с мостиковой подструктурой

В структуре системы выбирается базовый элемент и предполагается, что:

1. базовый элемент находится в полностью работоспособном состоянии;
2. базовый элемент находится в состоянии отказа.

Естественно, переменная, обозначающая этот элемент, принимает значения 1 или 0. И задача оценки надежности может быть сформулирована так:

$$\mathcal{P} = p_i M[\varphi(X, x_i = 1)] + (1 - p_i) M[\varphi(X, x_i = 0)].$$

Структурная функция надежности при расчете логико-вероятностным методом в этом случае будет иметь вид: $\varphi(BT) = (O7 \wedge (O1 \wedge (O8 \vee O9) \wedge ((O2 \vee O3) \wedge O5) \vee ((O4 \vee \overline{O4} \wedge RESET) \wedge O6)) \wedge FINISH)) \vee (\overline{O7} \wedge (O1 \wedge (((O2 \vee O3) \wedge O5 \wedge O9) \vee ((O4 \vee \overline{O4} \wedge RESET) \wedge O6 \wedge O8)) \wedge FINISH)$. Аналогичным образом усложняются расчеты с помощью других методов, а метод путей и сечений дает лишь приблизительные результаты. Тем не менее, расчет надежности подобной структуры с использованием ПКБТ не сложнее, чем расчет более простой схемы.