

На правах рукописи

Романников Дмитрий Олегович

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ С
ПРИМЕНЕНИЕМ UML ДИАГРАММ И СЕТЕЙ ПЕТРИ ДЛЯ
СИСТЕМ УПРАВЛЕНИЯ ЛОКАЛЬНЫМ ОБОРУДОВАНИЕМ

Специальность 05.13.11 – Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей

Автореферат диссертации на соискание ученой степени
кандидата технических наук

Новосибирск – 2012

Работа выполнена в Федеральном государственном бюджетном образовательном учреждении высшего профессионального образования «Новосибирский государственный технический университет»

Научный руководитель доктор технических наук, профессор
Воевода Александр Александрович

Официальные оппоненты: Малышкин Виктор Эммануилович, доктор технических наук, профессор, Федеральное государственное бюджетное учреждение науки Институт вычислительной математики и математической геофизики СО РАН, заведующий отделом математического обеспечения высокопроизводительных вычислительных систем

Задорожный Анатолий Филиппович, кандидат технических наук, доцент, Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования Новосибирский государственный архитектурно-строительный университет (Сибстрин), заведующий кафедрой информационных систем и технологий

Ведущая организация: Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Национальный исследовательский Томский политехнический университет», г. Томск

Защита состоится «20» сентября 2012 г. в 14.00 часов на заседании диссертационного совета Д 212.173.06 при ФГБОУ ВПО Новосибирском государственном техническом университете по адресу: 630092 РФ, г. Новосибирск, пр. К.Маркса 20.

С диссертацией можно ознакомиться в библиотеке ФГБОУ ВПО Новосибирского государственного технического университета.

Автореферат разослан «__» _____ 2012 г.

Ученый секретарь
диссертационного совета

Чубич Владимир Михайлович

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы. В настоящее время программное обеспечение (ПО) используется в большинстве сфер человеческой деятельности. От его надежности и функциональности зависит успешность работы многих предприятий. Поэтому качество разрабатываемого ПО приобретает важное значение. Существующие подходы к разработке ПО не гарантируют его качество в достаточной степени. Тестирование разрабатываемого продукта также не может гарантировать отсутствие ошибок в программе, так как многие из них могут проявляться в специфических условиях, которые невозможно смоделировать в тестовой среде. Таким образом, проблемы качества ПО при его разработке являются достаточно актуальными.

В настоящее время используются некоторые формальные способы разработки качественного ПО. Особый интерес представляют такие из них которые способны *гарантировать* качество и безотказность создаваемого ПО. Однако вышеприведенные способы разработки редко используются на практике из-за сложности их применения.

Одним из возможных способов формального доказательства верности ПО является применение алгоритмов на основе интеграции UML диаграмм и сетей Петри, математический аппарат которых достаточно подробно изучен В.Е.Котовым, Дж.Петерсоном и др., что позволяет использовать их для анализа модели системы (Д.А.Колесников, М.А.Ищенко, С.В.Коротиков, А.В.Доля, L.Baresi, M.Pezze, S.Bernardi, S.Donatelli, J.Merseguer, E.P.Naumovich).

Тем не менее, разработанные на настоящий момент методологические, технологические и инструментальные решения либо не позволяют выполнить моделирование для совокупности диаграмм и моделировать поведение нескольких объектов в системе одновременно, либо используют классические сети Петри, что в совокупности не дает возможность выполнить полное моделирование разрабатываемой системы. Следует также отметить, что рекомендации по анализу результатов моделирования сетей Петри нуждаются в дополнительном исследовании.

Цель работы. Разработать методику написания программного обеспечения с использованием UML диаграмм и сетей Петри, применимую для разработки систем управления локальным оборудованием.

Для достижения поставленной цели решаются следующие задачи:

- анализ современных подходов создания высококачественного программного обеспечения;
- определения ключевых особенностей алгоритмов разработки с использованием UML диаграмм и сетей Петри, подлежащих изменению или улучшению;
- разработка правил обнаружения программных ошибок обращения к несуществующему элементу массива и утечек памяти при работе с объектами на основе анализа результатов моделирования системы в сетях Петри;

– создание методики разработки ПО с использованием UML диаграмм и сетей Петри для применения к разработке ПО систем управления локальным оборудованием и обеспечения возможности моделирования систем с произвольным количеством объектов одного класса одновременно;

– применение методики разработки для решения задачи разработки программного обеспечения системы управления локальным оборудованием водонапорных станций водоканала. Проверить возможность применения предлагаемой методики для разработки ПО для задачи поиска пути движения манипулятора от начальной до конечной координат в замкнутом пространстве с препятствиями и разработке ресурсной задачи из игровой индустрии.

Объект исследования. Объектом исследования является методика разработки программного обеспечения, основанная на использовании UML диаграмм и сетей Петри.

Методы исследования. При решении задач диссертационного исследования использовались следующие методы: системный анализ; аппарат сетей Петри; разделы дискретной математики; объектно-ориентированный анализ.

При реализации примеров применения рассматриваемой методики использовались отраслевые и международные стандарты, CASE-технологии, современные инструментальные среды и пакеты моделирования.

Научная новизна. В диссертационном исследовании были достигнуты следующие результаты:

– методика разработки ПО с использованием диаграмм UML и сетей Петри для разработки ПО для систем локальной автоматизации. Изменен набор используемых UML диаграмм, что позволяет разрабатывать более сложные алгоритмы системы. Внесены изменения в набор правил формального преобразования диаграмм UML в цветные сети Петри, структура которых позволяет выполнять моделирование системы с несколькими объектами одного класса одновременно и моделирование статических методов класса;

– разработаны рекомендации по использованию UML диаграмм состояния и диаграмм действия для описания динамических свойств класса;

– предложены варианты обнаружения некоторых типов программных ошибок: обращение к несуществующим элементам массива, неверное выделение и освобождение памяти объектов – утечка памяти.

Практическая ценность и внедрение. Проведенные исследования подтвердили эффективность применения рассматриваемой методики на всех этапах разработки ПО.

Результаты диссертационной работы были использованы в разработке и внедрении ПО системы управления локальным оборудованием водонапорных станций в водоканале города Тюмень, в гранте Федерального Агентства по образованию ГК № П 694 от 12.08.2009 г. (конкурс НК-81П (2009 – 2011 гг.)), а также в конкурсе фундаментальных и прикладных исследований внутренних грантов НГТУ «Использование UML-диаграмм и аппарата сетей

Петри как формальных методик анализа архитектуры программного обеспечения». Результаты диссертационного исследования используются в учебном процессе НГТУ. Показана возможность применения не только для систем локальной автоматизации, но и для других задач, например: упрощенная задача автоматического поиска допустимых перемещений манипулятора в пространстве с препятствиями и разработка ресурсной задач игровой индустрии.

На защиту выносятся следующие положения:

– методика разработки ПО с использованием UML диаграмм и сетей Петри, включающая набор правил преобразования, позволяющих моделировать поведение нескольких объектов одного класса одновременно и выполнять моделирование статических классов системы;

– рекомендации по использованию UML диаграмм действия и состояния для описания динамических свойств класса;

– рекомендации по обнаружению программных ошибок обращения к несуществующим элементам массива и утечки памяти при работе с объектами;

– разработанная на основе методики разработки программного обеспечения с использованием UML диаграмм и сетей Петри программная система управления водонапорными станциями водоканала города Тюмень.

Апробация работы. Основные положения диссертации были изложены на следующих конференциях: международной научной заочной конференции «Актуальные вопросы современной техники и технологии» (Липецк, 24 апреля 2010 г.); XIII международной научно-технической конференции «Информационно-вычислительные технологии и их приложения» (Пенза, декабрь 2010 г.); международной научно-практической конференции «Традиции и инновации в современном социокультурном пространстве» (Новосибирск, 2011 г.); DST-RFBR Sponsored Indo-Russian Joint Workshop «Computational intelligence and modern heuristics in automation and robotics» (Surat, India, 20-22 September 2010); RFBR and DST sponsored The second Russian-Indian Joint Workshop «Computational intelligence and modern heuristics in automation and robotics» (Novosibirsk, Russia, 10-13 September 2011); XIII международной научно-практической конференции «Наука и современность» (Новосибирск, 2011 г.); международной заочной научно-практической конференции «Наука и техника XXI века» (Новосибирск, 2011 г.). А также в школе молодых ученых САИТ-2011 секции №2 «Информационные технологии в системах автоматического и автоматизированного управления» (Новосибирск, 12-16 сентября 2011 г.) и научных семинарах кафедры «Автоматика» НГТУ (2009 – 2011 гг.).

Публикации. Основные положения и результаты диссертационной работы опубликованы в 21 работах, в том числе: статей в изданиях, рекомендуемых ВАК РФ – 3; в сборниках научных трудов – 12, материалах международных симпозиумов и конференций – 6.

Структура и объём диссертации. Диссертация состоит из введения, четырех глав, заключения, библиографического списка использованной литературы и приложений. Работа изложена на 195 с. машинописного текста: основное содержание на 116 с. и включает 49 рисунков, 6 таблиц и список литературы из 106 наименований.

СОДЕРЖАНИЕ РАБОТЫ

Во **введении** рассмотрена актуальность разработки качественного программного обеспечения при помощи алгоритмов разработки с использованием UML диаграмм и сетей Петри. Определены цели и задачи диссертационного исследования, сформулированы положения, выносимые на защиту, их научная новизна и практическая ценность. Представлены основания для выполнения работы, ее апробация и структура.

В **первой главе** диссертации анализируются существующие подходы и алгоритмы к разработке ПО принятые в индустрии программирования. Среди всех наиболее подходящих для разработки качественного программного обеспечения подходов можно выделить итеративные методики. Для написания кода в индустрии программного обеспечения широко применяются принципы объектно-ориентированного программирования (ООП). В качестве языка графического отображения принципов ООП активно используются UML диаграммы, что значительно ускоряет процесс разработки ПО.

Однако только программные коды, в том числе через отражение в UML диаграммах, не могут гарантировать качества системы даже при высоком уровне тестирования, так как многие ошибки проявляются в специфических условиях, которые не воспроизводимы в тестовой среде. Для достижения качества программных систем выбираются сети Петри, позволяющие выполнять, как ручное моделирование системы (при помощи специальных программных пакетов), так и анализ пространства состояний системы для выявления вероятных ошибок в ПО. Анализ пространства состояния системы осуществляется при помощи анализа дерева достижимости сети Петри. Математический аппарат сетей Петри достаточно подробно изучен в последние 30-40 лет такими учеными как В. Е. Котов, Дж. Петерсон. Тем не менее, использование сетей Петри не нашло широкого применения на практике.

Сети Петри применяются во многих сферах разработки, например на рис. 1 представлена часть задачи о моделировании протокола связи между диспетчерской станцией и таксофоном. Протокол связи представлен сетью Петри при помощи пакета моделирования CPN Tools. На основании моделирования и анализа сети сделан вывод о наличии ошибок в системе: множественное ветвление сети и ее небезопасность. После введения дополнительных мест S1, S2 и S3, анализ сети проблем не выявил.

Отдельно стоит выделить методики разработки ПО с совместным использованием сетей Петри и UML диаграмм. Они обладают преимуществами математического аппарата сетей Петри, быстротой разработки и наглядностью представления UML диаграмм и с их помощью можно как оценивать

быстродействие разрабатываемых систем, так и применять для различного анализа ПО.

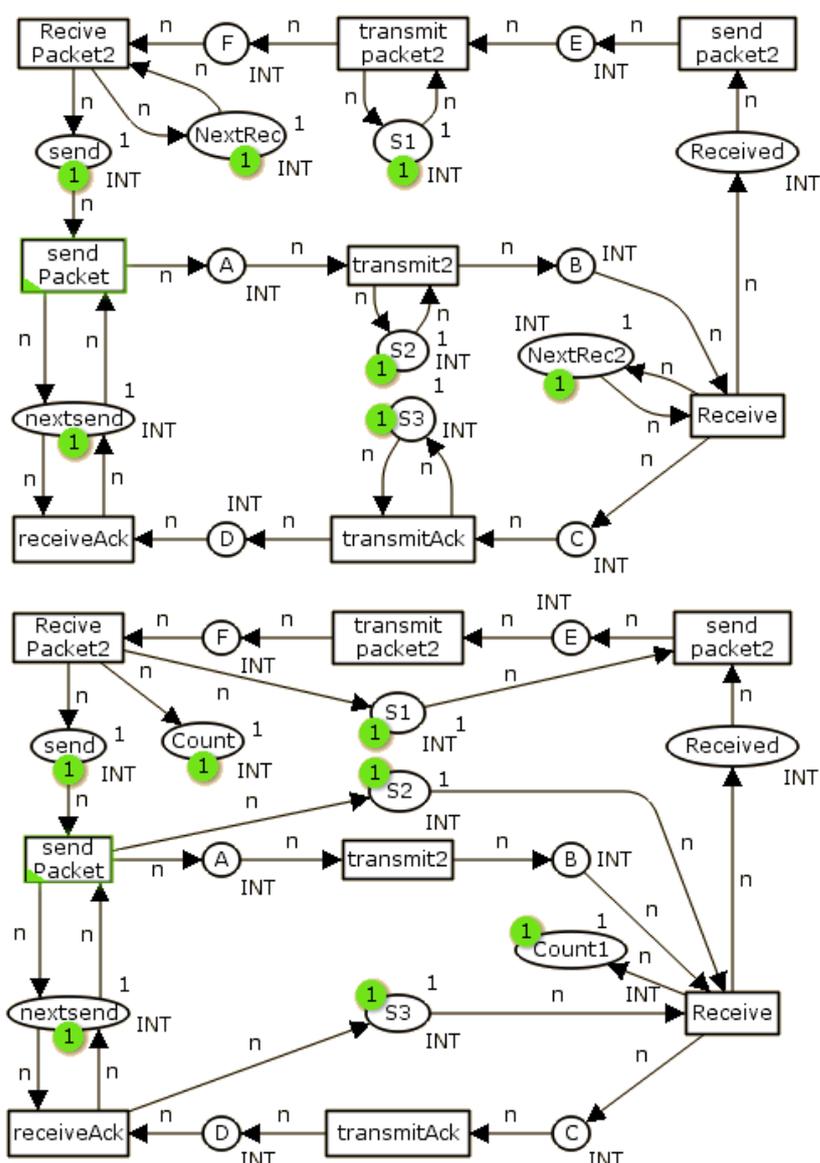


Рис. 1. Пример сети Петри: сверху с ошибкой, снизу – исправленная

Однако эти методики не свободны от недостатков, не позволяющих выполнить полное моделирование системы: отсутствие однозначного формального преобразования элементов UML диаграмм в сети Петри, отсутствие примеров использования и правил преобразования для таких инструментов UML диаграмм как (сохранение истории, стереотипы).

Также к недостаткам можно отнести такие проблемы анализа сети Петри, как: «взрыв» пространства состояний сети Петри, сворачиваемость сети Петри для идентичных объектов и отсутствие достаточно формальных методик анализа

ПО на основе результатов анализа модели, выполненной при помощи сетей Петри.

Во **второй** главе на основе комбинации и анализа наиболее удачных процедур, сформирован алгоритм разработки ПО, который позволит выявить их общие достоинства и недостатки. Приводятся рекомендации применения диаграмм состояний и действий для описания динамических свойств классов. Формулируются задачи диссертационного исследования.

В программной индустрии разработки ПО применяются различные процедуры с использованием UML диаграмм и сетей Петри, например в работах. Комбинируя и анализируя наиболее удачные процедуры, сформирован алгоритм разработки ПО, который позволяет выявить их общие достоинства и недостатки:

Рассмотрим данные недостатки:

– построение диаграммы состояний на основе сложных состояний, содержащих внутреннюю часть методов, не является правильным, так как в таком случае диаграмма состояний не отражает реальные состояния объекта, а показывает выполнение метода объекта в данный момент;

– использование диаграмм состояний не является обязательным, так как они могут быть заменены на другие диаграммы, которые описывают поведение системы, например, диаграммы действий;

– использование только диаграммы последовательности не способно отразить взаимодействие между различными классами в достаточной мере (диаграмма взаимодействий не позволяет выполнить построение сложных алгоритмов). Динамическое создание объектов, используемых в данном подходе, ограничивает описание межобъектного взаимодействия системы;

– представленный выше алгоритм разработки ПО является узкоспециализированным и направлен на создание ПО для систем с клиент-серверной архитектурой. Нет примеров его применения для создания систем другого класса;

– ручной прогон модели в сетях Петри по различным последовательностям выполнения является не обязательным и излишним. Так как ручной «прогон» системы в нотации сетей Петри не является целью разработки ПО и, по своей сути, не отличается от ручного «прогона» программы в программных кодах при помощи отладчика (*debugger*).

Рассмотрим диаграмму состояний (рис. 2) отображающую поведение заслонки. Изначально заслонка находится в положении *closed*. При срабатывании метода *on* состояние меняется на *opening* до того момента, как будет вызван метод *checkOn*. При его поступлении заслонка переходит в состояние *open*. Если сигнал не поступил в течение 50 секунд, то объект переходит в состояние *error*. Аналогичное поведение происходит при срабатывании метода *off*. Из состояния *error* в состояние *closed* задвижка переходит при вызове метода *ack*.

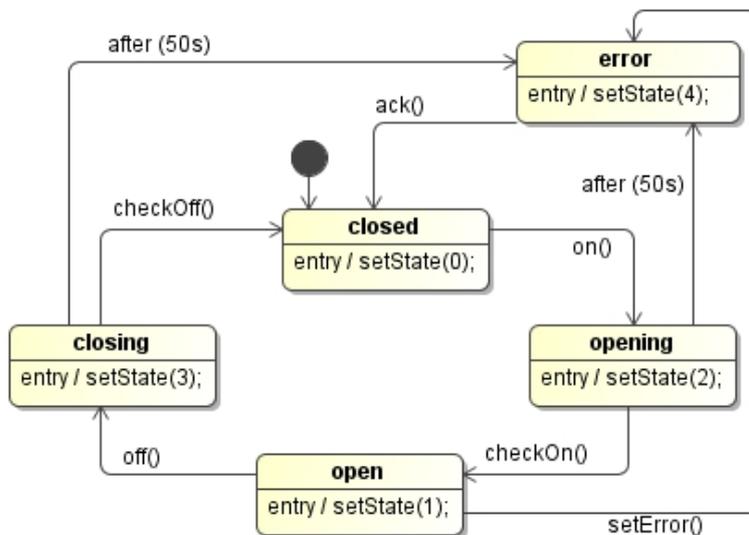


Рис. 2. Диаграмма состояний

Далее рассмотрим описание той же задачи при помощи диаграмм действий (рис. 3, 4). Диаграммы действий не имеют функционала для отображения состояний объекта, но при их помощи можно более детально и наглядно отобразить методы классов. Поэтому для описания динамических свойств класса используется описание ме-

тодов класса при помощи диаграмм действий. Вызов данных методов также осуществляется при помощи диаграммы отображающей межобъектное взаимодействие.

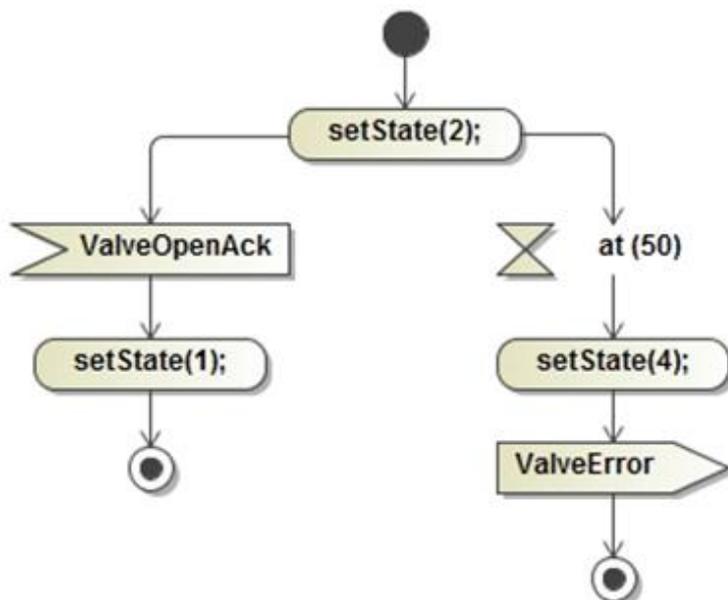


Рис. 3. Реализация метода *on* при помощи диаграмм действий

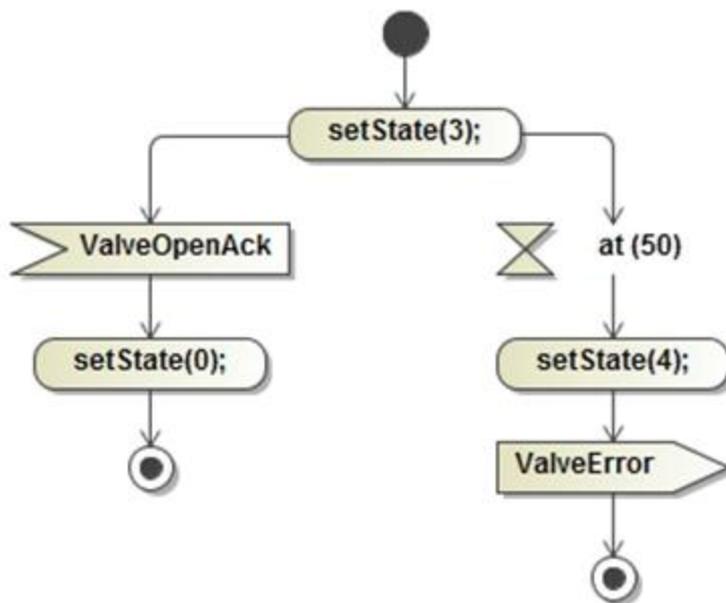


Рис. 4. Реализация метода *off* при помощи диаграмм действий

также при работе с внешними сигналами. При описании алгоритмической части ПО удобнее использовать диаграммы действий.

Рассмотрим сеть Петри (рис. 5), иллюстрирующую применение «старого» набора правил формального преобразования. Из данной сети видно, что при создании нескольких объектов класса *CThread1* посредством вызова сообщений *new1* и *new2*, метки попадают в места *WaitingForRec* и *Working*. При срабатывании события *dell*, моделирующего удаление объекта, в сети к

Приведем некоторые рекомендации по использованию UML диаграмм состояния и действия для описания динамических свойств классов. Использование диаграмм действий обосновано при линейном исполнении программы и описании алгоритмической части ПО. Использование диаграмм действий является более наглядным при наличии малого количества состояния объекта, так как при этом диаграмма состояний будет представлять одно состояние с множеством переходов, замкнутых на данное состояние. Возможным дальнейшим развитием UML диаграмм будет являться объединение диаграмм действий и диаграмм состояния. Рекомендовано использовать диаграммы состояний при наличии множества состояний у класса, а

удалению будут доступны сразу два существующих объекта, а не только первый объект как следует из логики программы.

Таким образом, сформулируем недостатки правил формального преобразования. К основным относятся *неприспособленность сети Петри для моделирования (создания) двух и более объектов одного класса*, отсутствие возможности получения результата выполнения метода и моделирования статических методов.

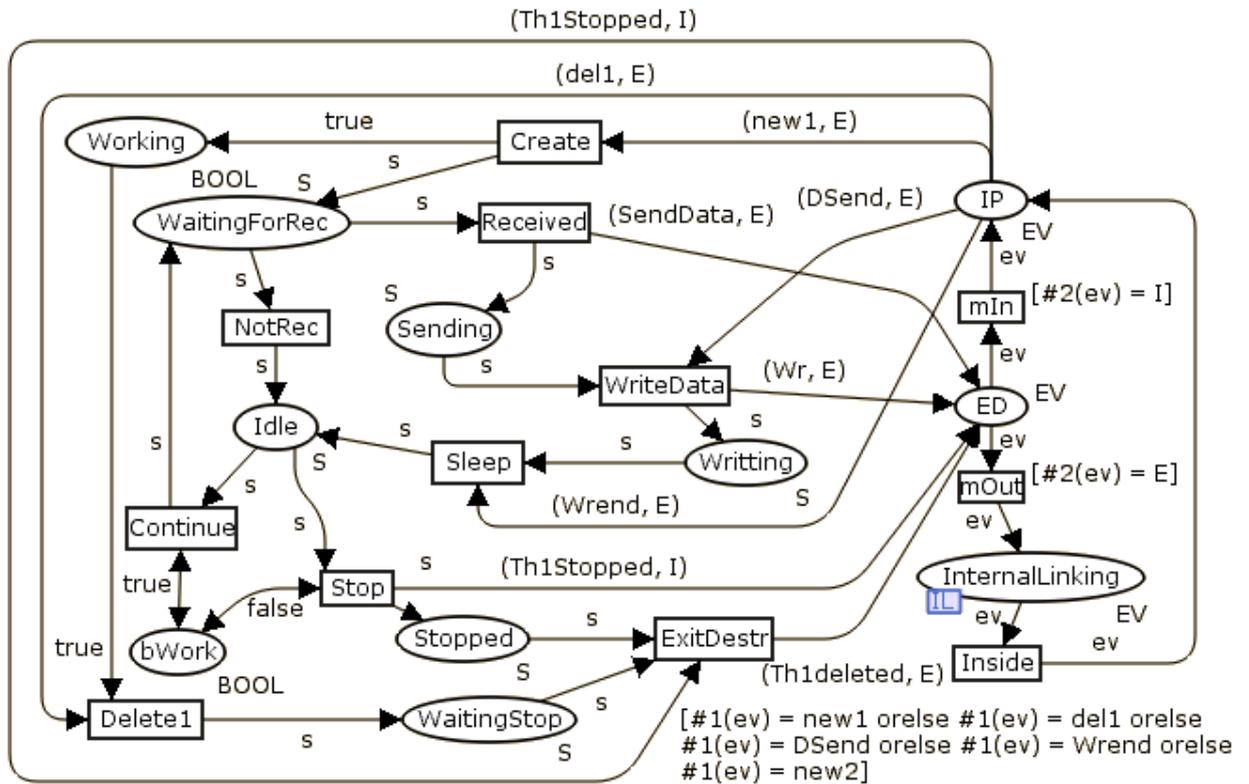


Рис. 5. Пример сети Петри для иллюстрации «старого» набора правил преобразования

Следует отметить, что в опубликованных работах в основном рассматриваются правила и иллюстрации данных правил преобразования одиночных диаграмм или наборов диаграмм в сети Петри. Есть ряд работ по анализу UML диаграмм в вопросах производительности. Однако данные доводы не являются ключевыми для широкого использования рассматриваемой методики в программной инженерии. Существенным аспектом является возможность обнаружения и указания вероятных программных ошибок уже на этапе разработки программы.

Вторая глава заканчивается постановкой задач диссертационного исследования для устранения вышеприведенных недостатков разработки и создания качественного ПО с использованием UML диаграмм и сетей Петри.

Третья глава посвящена вопросам конструирования методики разработки ПО для систем локальной автоматки, с учетом устранения приведен-

ных недостатков правил преобразования и выявленных ошибок в исходной программе на основе анализа результатов моделирования сетей Петри.

Во второй главе выделены существенные недостатки используемого алгоритма разработки ПО, среди которых наиболее существенными являются: отсутствие возможности моделирования поведения нескольких объектов одновременно, неприспособленность сети Петри к моделированию статических методов и невозможность получения результата выполнения метода. Следствием вышеприведенных недостатков является неполноценное моделирование системы и вероятность потенциального пропуска ошибок моделирования.

Предложены *изменения* в правилах формирования сети Петри на основе UML диаграмм:

- к переходам сети необходимо добавить защитные условия, которые будут определять, какая метка (метка представляет собой объект) может совершить переход (т.е. для какого объекта будет вызван метод). Защитное условие будет работать на основе уникального идентификатора объекта, к которому может относиться как имя объекта, так и обычный целочисленный номер. Идентификатор объекта должен храниться в метке, которая моделирует его поведение. Для этого необходимо расширить тип метки и ввести дополнительное значение для хранения идентификатора;

- для вызова методов использовать дополнительные места, соединенные с переходом, обозначающим срабатывание метода;

- возврат программы на место вызова метода и возвращение значения выполнения метода осуществлять при помощи дополнительного места, соединенного с переходом, в котором заканчивается выполнение метода;

- для моделирования статических методов использовать такую же схему, как и для моделирования обычных методов, за исключением того, что переходы и места такого метода не соединены с местами и объектами самого метода.

Сформированная с помощью предлагаемых изменений в правилах преобразования сеть Петри представлена на рис. 6.

В дополнение к вышеприведенным правилам преобразования необходимо определить тип места для моделирования вызова методов класса в сетях Петри. Тип места определяется входными параметрами метода в UML диаграмме классов. Кроме параметров метода необходимо передать дополнительные «служебные» параметры, которые требуются для правильного согласования между участками сети Петри. К таким параметрам относятся: идентификатор объекта, для которого вызван данный метод, и идентификатор потока, в котором произошел вызов метода. Отсюда, общая структура типа данных метки будет иметь следующий вид: $(ID, OPTIONS)$, где ID – идентификатор объекта и потока, а $OPTIONS$ – параметры, передающиеся в метод. Идентификатор объекта и потока представляет собой пару ID_O*ID_T , где первый – идентификатор объекта, а второй – потока. Для идентификации объектов и потоков можно использовать любые уникальные

идентификаторы, например в данной работе использовались целочисленные значения.

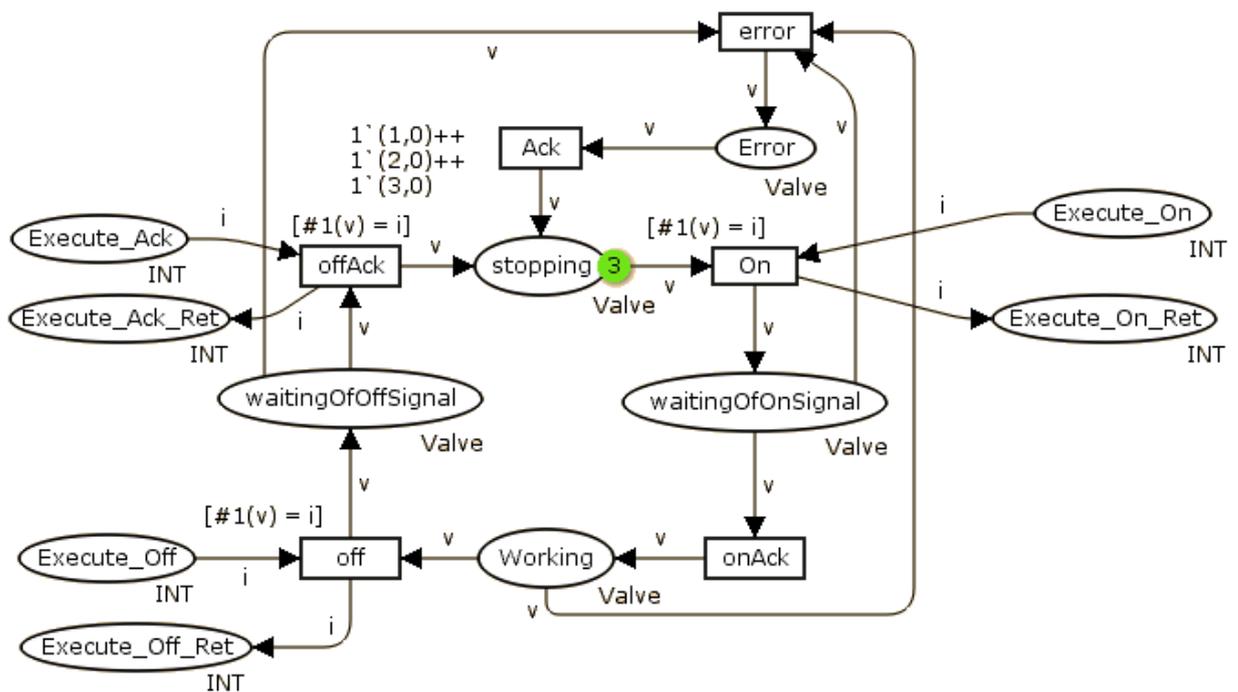


Рис. 6. Сеть Петри, сформированная с помощью модифицированных правил формального преобразования

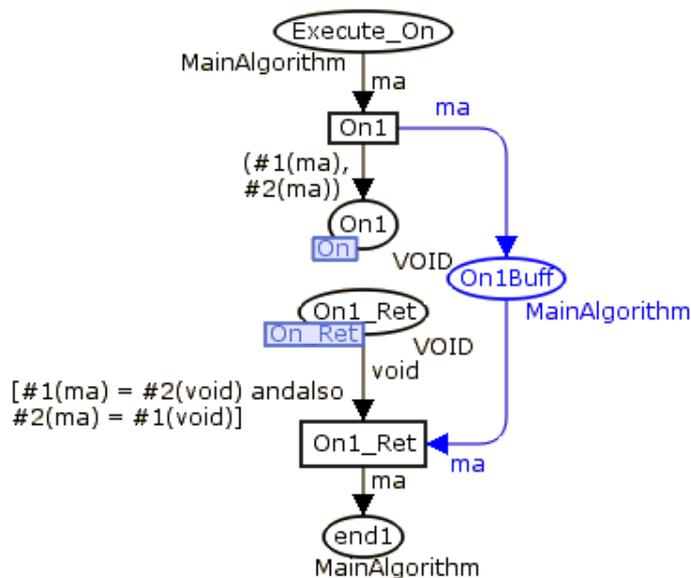


Рис. 7. Вызов метода On класса Valve

перехода *On1_Ret* и мест *On1_Ret* и *On1Buff*. При вызове метода метка попадает в место *On1Buff*. Когда метод закончил свое выполнение, то метка переходит в место *On1_Ret*. Таким образом, переход *On1_Ret* становится доступен при выполнении защитного условия $[\#1(ma) = \#2(void) \text{ andalso } \#2(ma) = \#1(void)]$, которое определяет, в какой участок сети должна быть возвращена метка.

На рис. 7 приведен пример вызова метода *On* класса *Valve*, изображенного на рис. 6. Вызов метода *On* осуществляется при помощи мест *On1* и *On1_Ret*. Для моделирования вызова метода необходимо, чтобы сработал переход *On1*, при этом одна метка с идентификатором и параметрами переместится в место *On1*, а вторая метка, в которой будут сохраняться значения до вызова метода, переходит в место *On1Buff*.

Выход из метода осуществляется при помощи пе-

В третьей главе рассмотрены вопросы анализа результатов моделирования сети Петри и сформированы правила обнаружения программных ошибок обращения к несуществующему элементу массива и утечки памяти при работе с объектами.

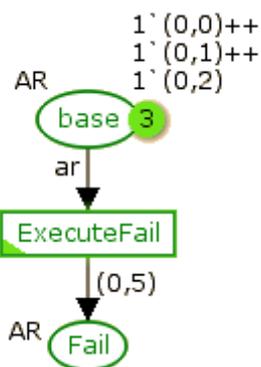


Рис. 8. Сеть Петри

Рассмотрим сеть Петри (рис. 8), отображающую программу, в которой инициализируется массив размерностью в три элемента, а затем выполняется обращение к пятому элементу массива, т.е. к несуществующему элементу. Данную ошибку можно обнаружить следующими способами: при ручном моделировании системы (не является приоритетным способом); при анализе сети Петри на основе пространства состояний с ограниченным количеством элементов массива в соответствии с кодом программы; при анализе сети Петри на основе пространства состояний с неограни-

ченным количеством элементов массива. В данном случае моделирование пройдет успешно, но учитывая свойства ограниченности сети и статистику, полученную из отчета анализа состояний, можно сделать вывод о наличии ошибки. Место *Fail* на сети *execution* содержит три метки с максимальным значением индекса массива «5» – «*execution'Fail 1 3 (0,5)*»

Рассмотрим сеть Петри (рис. 9 и 10). Она представляет собой класс *memory_leak* (рис. 9) и основную часть программы (рис. 10), содержащую последовательный вызов десяти конструкторов и одного деструктора, что приводит к ошибке утечки памяти. Обнаружение ошибки утечки памяти основано на анализе результатов моделирования (таблица 1).

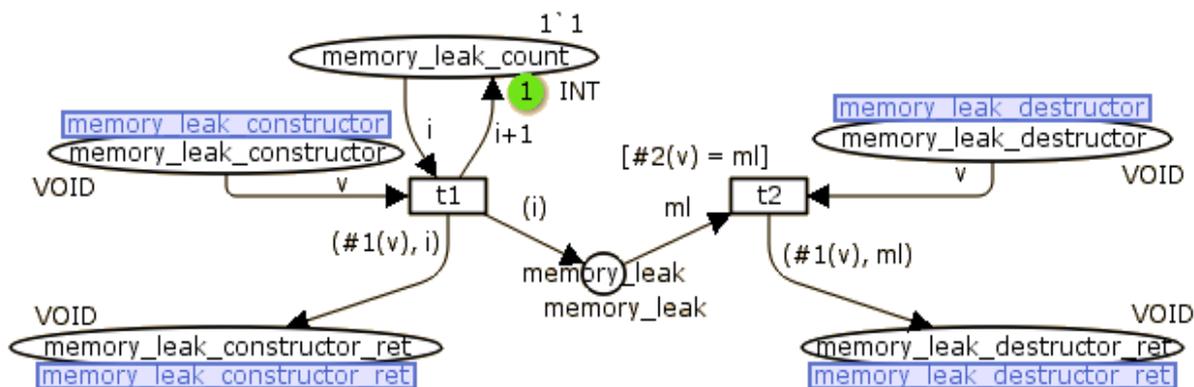


Рис. 9. Сеть Петри для класса *memory_leak*

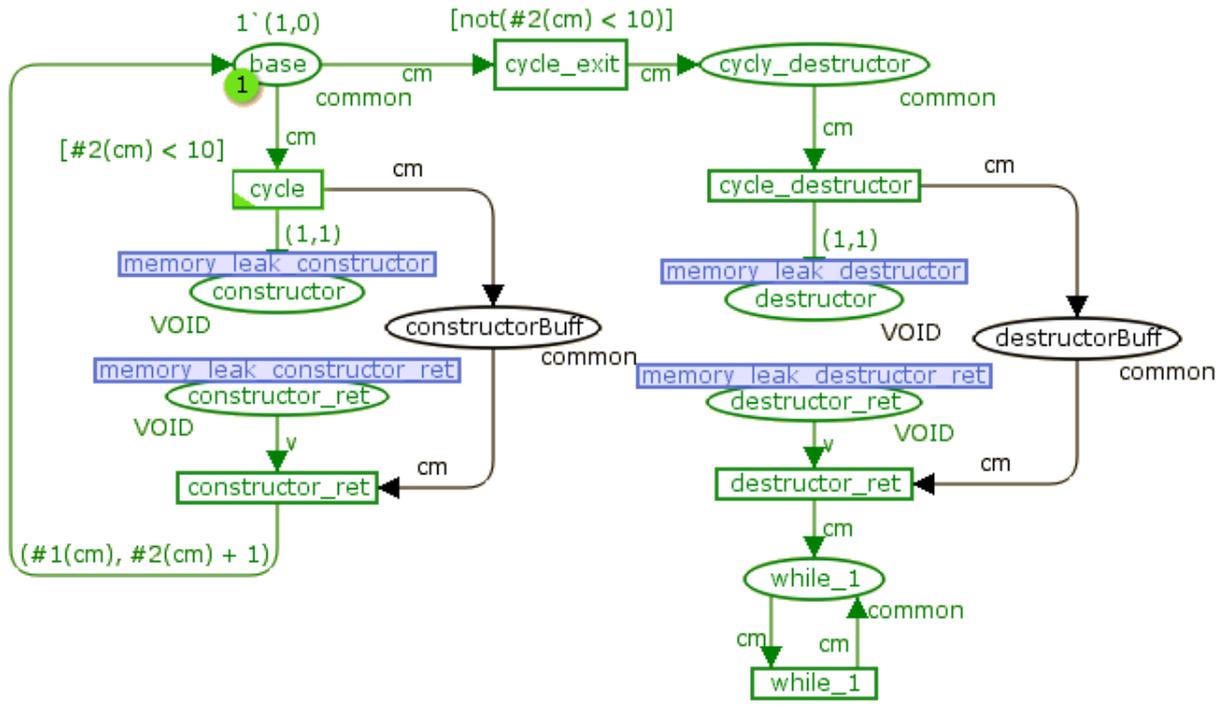


Рис. 10. Сеть Петри для программы main

Таблица 1

Результаты анализа сети Петри

Boundedness Properties ¹	
Best Upper Multi-set Bounds	memory_leak'memory_leak_constructor 1
execution'constructor_ret 1	1^(1,1)
1^(1,1)++	memory_leak'memory_leak_count 1
1^(1,2)++	1^1++
1^(1,3)++	1^2++
1^(1,4)++	1^3++
1^(1,5)++	1^4++
1^(1,6)++	1^5++
1^(1,7)++	1^6++
1^(1,8)++	1^7++
1^(1,9)++	1^8++
1^(1,10)	1^9++
execution'constructor 1	1^10++
1^(1,1)	1^11
execution'cycly_destructor 1	memory_leak'memory_leak_destructor 1
1^(1,10)	1^(1,1)
execution'destructor 1	memory_leak'memory_leak_destructor_ret 1
1^(1,1)	1^(1,1)

Из таблицы видно, что обращение к конструктору (*execution'constructor_ret*) и деструктору (*execution'destructor_ret*) выполня-

¹ В отчете приведены не полный набор свойства ограниченности сети, а только те свойства, которые требуются для определения ошибки.

лось разное количество раз, что указывает на наличие ошибки утечки памяти. По схожему принципу можно определить утечки памяти при использовании функций семейства *malloc* и *free*.

Третья глава заканчивается формулировкой методики разработки программного обеспечения с использованием UML диаграмм и сетей Петри для систем управления локальным оборудованием. **Предлагаемая методика** состоит из последовательно выполняемых шагов. Рассмотрим их.

Шаг 1. Формирование требований к разрабатываемой системе при помощи построения *диаграмм вариантов использования*. На первом шаге выделяются используемые в системе актеры, прецеденты и варианты использования системы.

Шаг 2. Составление *диаграммы классов*. Выделяются сущности системы, для которых составляются классы или иерархия классов в системе. На втором шаге необходимо выделить методы классов.

Шаг 3. Следующим шагом является составление диаграмм для описания динамических свойств системы. Для выделенных классов составляются *диаграммы состояний или действий*. Смена состояний достигается за счет обмена сигналами между объектами. Диаграммы действий необходимо составить для каждого метода из диаграммы классов.

Шаг 4. Четвертый шаг – составление *диаграммы объектов*. На основании диаграммы классов выделяются объекты, которые будут участвовать в системе. Диаграмма объектов определяет начальные значения объектов системы, а также значение меток для сети Петри.

Шаг 5. Далее составляем диаграммы основного алгоритма системы. Данные диаграммы должны отображать динамические свойства системы. Как и в третьем шаге для этих целей могут быть использованы *диаграммы состояний или действий*.

Шаг 6. На этом шаге осуществляется *преобразование* составленных UML диаграмм состояний и действий в сеть Петри при помощи правил формального преобразования.

Шаг 7. Анализ полученной сети Петри. Выполняется при помощи программного пакета моделирования CPN Tools или любого другого автоматизированного программного пакета для работы с сетями Петри. Анализ позволяет определять имеющиеся ошибки и, в зависимости от их типа, вносить изменения в набор UML диаграмм на необходимых этапах методики. В случае отсутствия ошибок составленный набор UML диаграмм можно считать верным и приступать к генерации исполняемого кода.

В *четвертой главе* продемонстрировано применение результатов, полученных в главах 2, 3 при разработке системы управления водонапорными станциями водоканала.

Применение, предложенных в третьей главе методики разработки, правил формального преобразования, позволило разработать алгоритмы работы водонапорных станций в виде набора UML диаграмм, преобразовать полученный набор в сети Петри и выполнить анализ на различные аспекты. На

первом этапе предлагаемой методики разработки, на основании словесного описания поставленной задачи составлены диаграммы вариантов использования системы, которые отображают наиболее значимые функции. На втором этапе формируются диаграммы классов системы. Третий этап включает разработку диаграмм состояния для описания динамических свойств классов. На четвертом этапе составляется диаграмма действий, отображающая основной алгоритм системы. Содержанием пятого этапа является преобразование составленного набора UML диаграмм в сети Петри. Непосредственное моделирование и анализ работы системы в части управления локальным оборудованием с помощью сети Петри, в ходе анализа которой было построено пространство состояний, содержащее более 157 тысяч узлов и более 50 тысяч дуг, позволило тщательно исследовать все особенности функционирования разрабатываемой системы и выявить ряд логических и программных ошибок.

В приложениях приведены акты внедрения программной системы, разработанной на основании предложенной методики разработки и внедрения результатов диссертационного исследования в учебный процесс. Также приведены примеры применения рассматриваемой методики в задачах из других программных областей. Применение предлагаемой методики рассматривается на задаче перемещения манипулятора в пространстве с препятствиями, которая решается перебором возможных вариантов перемещения манипулятора. Другой задачей, к которой применяется предлагаемая методика, является задача индустрии программирования алгоритмов управления ресурсами и изменения положения в пространстве игровых моделей из области игровой индустрии.

ЗАКЛЮЧЕНИЕ

В заключении подводятся итоги проделанной работе, формулируются основные результаты анализа и развития современных подходов индустрии программного обеспечения к разработке высококачественного ПО.

В приложениях приведены акты внедрения программной системы, разработанной на основании предложенной модифицированной методики разработки и внедрения лабораторной работы в учебный процесс. Также приведены примеры применения рассматриваемой методики в задачах различной направленности: при рассмотрении движения манипулятора в замкнутом пространстве с препятствиями и разработки ресурсных задач игровой индустрии (на примере игры «змейка»).

Разработка программного обеспечения в настоящее время является актуальной задачей, для решения которой следует применять методики, способные гарантировать стабильную работу разрабатываемого кода при различных условиях эксплуатации. Актуальность проблемы подчеркивает и важность максимальной автоматизации предлагаемых методик, связанных с необходимостью их использования в реальных проектах, где критичным становится не только время разработки, но и быстрое выявление и устранение возможных ошибок. В соответствии с поставленными в диссертационной работе задачами были получены следующие результаты:

– предложена методика разработки программного обеспечения, основанная на использовании UML диаграмм и сетей Петри, позволяющая разрабатывать ПО для систем управления локальной автоматикой. В предложенной методике используется измененный набор UML диаграмм, который позволяет разрабатывать более сложные программные алгоритмы. Кроме того, выполнены изменения в правилах формального преобразования UML диаграмм в цветные сети Петри, при помощи которых формируется структура сети. Измененная структура сети Петри позволяет моделировать поведение нескольких объектов одного класса одновременно и выполнять моделирование статических методов классов, что в совокупности приводит к полному моделированию системы и ее анализу;

– предложен вариант обнаружения программных ошибок утечки памяти и обращение к несуществующим элементам массива на основании результатов моделирования сети Петри;

– приведены рекомендации по использованию UML диаграмм действий и состояний для описания динамических свойств класса.

В результате применения рассматриваемой методики разработчик получает возможность анализа разрабатываемого ПО, в котором используется более одного объекта класса одновременно, и его проверки на наличие программных ошибок. Предлагаемая методика была применена к задаче разработки программного обеспечения для системы управления локальным оборудованием водонапорных станций водоканала г. Тюмени. Система успешно прошла опытно-промышленную эксплуатацию и была сдана в промышленную эксплуатацию в 2011 г. Результаты диссертационных исследований используются в учебном процессе на факультете «Автоматика и вычислительная техника» НГТУ. Также методика была опробована на задачах других классов:

– задачи разработки ПО перемещения манипулятора в ограниченном пространстве с препятствиями на примере упрощенной задачи перемещения точки;

– ресурсной задачи из игровой индустрии (упрощенный вариант – игра «змейка»).

Основные публикации по теме диссертации

Статьи в журналах, рекомендованных ВАК для публикации результатов диссертаций на соискание ученой степени доктора и кандидата наук:

1. А.А. Воевода, Д.О. Романников, И.В. Зимаев. Применение UML диаграмм и сетей Петри при разработке встраиваемого программного обеспечения // Науч. вестн. НГТУ. - 2009. – №4 (37). - с. 169-174.

2. А.А. Воевода, Д.О. Романников. Редуцирование пространства состояний сети Петри для объектов из одного класса // Науч. вестн. НГТУ. - 2011. - №4 (45). - с. 146 – 150.

3. Д.О. Романников, А.В. Марков. Пример применения методики разработки ПО с использованием UML диаграмм и сетей Петри // Науч. вестн. НГТУ. - 2012. - №1 (46). - с. 175 – 181.

Другие издания:

4. А.А. Воевода, Д.О. Романников. Использование UML-диаграмм и временных сетей Петри в методе разработки ПО ч.2 // Сб. науч. тр. НГТУ. - 2010. - №4 (62).- с. 117-126.

5. А.А. Воевода, Д.О. Романников. Временные сети Петри и диаграммы UML // Сб. науч. тр. НГТУ. - 2010. - №1 (59). – с. 79-84.

6. А.А. Воевода, Д.О. Романников. О проектировании программного обеспечения для микроконтроллера с использованием UML // Сб. науч. тр. НГТУ. - 2009. - №4 (58). - с. 35-40.

7. А.А. Воевода, Д.О. Романников. Моделирование сетей Петри в CPN TOOLS // Сб. науч. тр. НГТУ. - 2008. - №3 (53). – с. 49-54.

8. А.А. Воевода, Д.О. Романников. О компактном представлении языков раскрашенных сетей Петри // Сб. науч. тр. НГТУ. - 2008. - №3 (53). – с. 105-108.

9. А.А. Воевода, Д.О. Романников. О моделировании систем реального времени с использованием UML и сетей Петри // Сб. науч. тр. НГТУ. - 2009. - №1 (55). – с. 63-66.

10. А.А. Воевода, Д.О. Романников. Особенности проектирования систем реального времени при по-мощи UML и сетей Петри // Сб. науч. тр. НГТУ. - 2009. - №1 (55). – с. 57-62.

11. А.А. Воевода, Д.О. Романников. О преобразовании UML диаграмм в сети Петри // Сборник статей XIV Международной научно-технической конференции «Информационно-вычислительные технологии и их приложения». – 2010. – с. 60 – 64.

12. А.А. Воевода, Д.О. Романников. Разработка ПО встраиваемых систем с использованием UML диаграмм и сетей Петри // Сборник докладов международной научной заочной конференции «Актуальные вопросы современной техники и технологии». – 2010. – с. 24 – 28.

13. Д.О. Романников. Роль автоматизации программирования в инновационных процессах // Труды международной научно-практической конференции «Традиции и инновации в современном социокультурном пространстве». – 2010. – с. 60 – 61.

14. А.В. Марков, Д.О. Романников, А.А. Воевода. Моделирование процесса поиска пути в лабиринте с помощью UML диаграмм и сетей Петри // Материалы международной заочной научно-практической конференции «Наука и техника XXI века». – 2011. – с. 84 – 89.

15. А.В. Марков, Д.О. Романников. Совокупное использование сетей Петри и UML-диаграмм при разработке программного обеспечения. // Сб. науч. тр. НГТУ. - 2011. - №2 (64). - с. 85–94.

16. Д.О. Романников, А.В. Марков, И.В. Зимаев. Обзор работ, посвященных разработке ПО с использованием UML и сетей Петри // Сб. науч. тр. НГТУ. - 2011. - №1 (63). - с. 91-104.

17. Д.О. Романников. Перспективы развития методики разработки программного обеспечения с использованием UML-диаграмм и сетей Петри // Сб. науч. тр. НГТУ. - 2010. - №2 (60). – с. 181 – 184.

18. Д.О. Романников. Применение методики разработки ПО на примере разработки игры «Змейка» // Сб. науч. тр. НГТУ. - 2011. - №4 (66). – с. 91 – 108.

19. A.A. Voevoda, D.O. Romannikov. Using timed Petri nets in approach of software design with UML diagrams // Proceedings on DST-RFBR Sponsored Indo-Russian Joint Workshop on «Computational intelligence and modern heuristics in automation and robotics». – 20 – 22 September 2010. Surat, India. – p. 95 – 97 [Использование временных сетей Петри в методе разработке программного обеспечения с использованием UML диаграмм].

20. A.A. Voevoda, A.V. Markov, D.O. Romannikov. Petri Nets modeling with supporting of system history storing // Proceedings on RFBR and DST sponsored The second Russian-Indian Joint Workshop «Computational intelligence and modern heuristics in automation and robotics». – 10 – 13 September 2011. Novosibirsk, Russia. – p. 47 – 49 [Моделирования сетей Петри с поддержкой сохранения системной истории].

21. Д.О. Романников. Нахождение ошибок обращения к несуществующим элементам массива на основании результатов анализа сети Петри // Сб. науч. тр. НГТУ. - 2012. - №1 (67). - с. 115-120.

Отпечатано в типографии Новосибирского
государственного технического университета
630092, г. Новосибирск, пр. К.Маркса, 20, тел./факс: (383)346-08-57
формат 60x84 1/16, объем 1,25 п.л., тираж 100 экз.
заказ № 1013 подписано в печать 25.06.2012 г.