

На правах рукописи

Аунг Аунг Хейн

**МОДЕЛИРОВАНИЕ И РАЗРАБОТКА СЕРВИС-ОРИЕНТИРОВАННЫХ
ПРИЛОЖЕНИЙ**

05.13.11 – математическое и программное обеспечение вычислительных
машин, комплексов и компьютерных сетей

АВТОРЕФЕРАТ

диссертации на соискание ученой степени
кандидата технических наук



Автор: _____

Москва – 2013

Работа выполнена в Национальном исследовательском ядерном университете «МИФИ»

Научный руководитель: доктор технических наук, профессор
Щукин Борис Алексеевич

Официальные оппоненты: Торхов Валерий Леонидович
доктор технических наук, профессор,
председатель правления ОАО АКБ
«Авангард», член комитета Ассоциации
Российских Банков по информационным
и интернет технологиям

Шевчук Сергей Сергеевич
кандидат технических наук,
главный специалист департамента
информационных технологий ОАО
«Ингосстрах»

Ведущая организация: Федеральное государственное бюджетное
образовательное учреждение Московский
Государственный Индустриальный
Университет (МГИУ)

Защита состоится «06» марта 2013 г. в 15 час. 00 мин. на заседании диссертационного совета Д.212.130.03 при Национальном исследовательском ядерном университете «МИФИ», расположенном по адресу: 115409, г. Москва, Каширское шоссе, 31.

С диссертацией можно ознакомиться в библиотеке Национального исследовательского ядерного университета «МИФИ».

Автореферат разослан «05» февраля 2013 г.

Отзывы и замечания по автореферату в двух экземплярах, заверенные печатью, просьба высылать по вышеуказанному адресу на имя ученого секретаря диссертационного совета.

Ученый секретарь
диссертационного совета
д.т.н., доцент

 Леонова М.Н.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы. В последнее время становится популярной покомпонентная сборка приложений на базе сервис-ориентированной архитектуры (SOA, Service-Oriented Architecture). Концепция SOA основана на том, что бизнес-приложение состоит из большого количества компонентов. Каждый такой компонент представляет собой сервис, являющийся модулем системы SOA. Использование сервис-ориентированного подхода позволяет крупному предприятию получить преимущества при дальнейшем развитии своей IT-инфраструктуры.

Для создания сервис-ориентированных приложений, в частности, перспективно использование технологии веб-сервисов. Технология веб-сервисов позволяет приложениям взаимодействовать друг с другом независимо от платформы, на которой они развернуты, а также от языка программирования, на котором они написаны. У веб-сервиса существует программный интерфейс, который описывает набор операций (веб-методов), удаленно вызываемых посредством стандартизованных XML сообщений. В отличие от традиционного веб-приложения, веб-сервис не обслуживает конечных пользователей, так как у него нет пользовательского интерфейса. Его главной задачей является предоставление услуг другим приложениям, таким как веб-приложения, мобильные или консольные приложения.

Ориентация на закрытые стандарты и форматы при разработке информационных систем создает проблемы при интеграции приложений. Веб-сервисы получили широкое распространение, так как они основаны на открытых протоколах обмена и передачи данных и стандартах, поддерживаемых на различных программных платформах. Это создает основу как для использования разработанных сервисов в рамках IT-инфраструктуры предприятия, так и для автоматизации бизнес процессов, в реализации которых участвуют несколько предприятий.

Построение информационных систем на базе веб-сервисов привело к понятию архитектуры, ориентированной на веб-сервисы (Web Services Architecture - WSA), разновидности сервис-ориентированной архитектуры. В диссертации исследуются вопросы разработки и моделирования веб-сервисов, используемых в рамках WSA, поэтому в дальнейшем под словом «сервис» понимается именно веб-сервис.

Сервис-ориентированные приложения – это распределенные приложения, функциональные компоненты которых – веб сервисы - выполняются на разных серверах и образуют «оркестр», управляемый клиентом. В силу этого они работают в значительной степени в условиях неопределенности. В связи с этим, в последнее время появилось большое количество исследований, связанных с построением моделей сервис-ориентированных приложений, выполненных с использованием сетей Петри.

Следует отметить работы К. Бохана и М. Худолея, посвященные построению моделей корпоративных приложений на базе иерархических сетей Петри, работы А. Martens, R. Hamadi, B. Benatallah, P. Xiong, Y. Fan, M. Zhou, связанные с моделями композитных сервисов, и множество других.

В настоящее время, в связи с быстрым распространением мобильных устройств весьма актуальны мобильные приложения, выполняющие самую разнообразную работу. В основном мобильные приложения предназначены для взаимодействия с корпоративной базой данных. Организовать такое взаимодействие легче всего, используя веб-сервис и сеть интернет для транспорта сообщений. При разработке мобильных приложений, как показывает практика, весьма актуально предварительное моделирование, позволяющее оценить как архитектурные решения, так и особенности функционирования мобильного приложения в реальных условиях.

Целью диссертационной работы является повышение эффективности разработки веб-сервисов и систем, ориентированных на применение веб-сервисов, на основе исследования существующих методов и инструментальных программных средств разработки и моделирования, используемых при композиции веб-сервисов и создании мобильных приложений.

Объектами исследований диссертационной работы являются веб-сервисы, как простые, так и композитные, а также приложения, построенные на их основе.

Предметами исследований диссертационной работы являются методы и средства разработки и композиции веб-сервисов.

Методами исследований диссертационной работы являются методы теории сетей Петри.

Научная новизна диссертации заключается в следующем:

- Разработаны модели веб-сервисов и их композиций на базе простых и раскрашенных сетей Петри

- Разработаны модели статического и динамического обращения к сервису.
- Предложен и продемонстрирован метод разработки веб-сервисов «сверху-вниз», в процессе которого сначала разрабатывается синтаксическое описание сервиса, а затем выполняется кодирование его методов.

Практическая значимость диссертации

- Разработанные методы и модели могут быть использованы при разработке веб-сервисов, как простых, так и композитных;
- Проведенный анализ методов разработки веб-сервисов и модели композиции, созданные на базе сетей Петри, могут эффективно использоваться в учебном процессе;

Основные положения, выносимые на защиту

- Предлагаемый подход к моделированию веб-сервисов и мобильных приложений, построенных с их использованием.
- Модели веб-сервисов и мобильных приложений, построенных с их использованием.

Апробация работы. Основные результаты диссертации докладывались и обсуждались на:

- XIX международном научно-техническом семинаре «Современные технологии в задачах управления, автоматизации и обработки информации». Алушта, 2010
- XIV международной телекоммуникационной конференции студентов и молодых ученых «Молодежь и наука». Научная сессия НИЯУ МИФИ-2011
- VIII Курчатовской молодежной научной школе, Москва, 2011 г
- XX международном научно-техническом семинаре «Современные технологии в задачах управления, автоматизации и обработки информации». Алушта, 2011

Внедрение результатов исследований

Работа по теме диссертации велась в сотрудничестве с сервисным центром компании БПЦ (www.bpc.ru). Методы и модели, представленные в диссертации, использованы при разработке:

1. Системы моделирования процессов инкассации банкоматов, выполненной в виде композиции 4 веб-сервисов и использованной в качестве прототипа при разработке промышленного варианта;
2. Системы, состоящей из веб-сервиса, работающего с офисной базой данных, и взаимодействующего с ним мобильного приложения, позволяющего планировать, оперативно отслеживать выполнение и изменять задания сотрудников, выполняемых вне офиса. Система внедрена в промышленную эксплуатацию и получен акт об использовании результатов диссертации.

Публикации

По теме диссертации опубликовано 7 печатных работ в период с 2010 по 2012 г, в том числе 3 в российских периодических изданиях, рекомендованных ВАК, тезисы 4 докладов включены в сборники трудов научно-технических конференций. Основные положения диссертационной работы отражены в опубликованных соискателем результатах.

Структура и объем диссертации

Диссертация состоит из введения, четырех глав, заключения и списка литературы и приложений. Общий объем основного текста, без учета приложений — 120 страниц, с учетом приложений — 150. Диссертация содержит 62 рисунка, 4 таблицы и 34 листинга программного кода. Список литературы включает 63 источника.

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Во введении обоснована актуальность темы диссертации, её научная новизна и практическая значимость, сформулирована цель работы.

В первой главе рассматриваются основные положения технологии веб-сервисов. В конце первого раздела диссертации поставлена цель и конкретные задачи диссертационного исследования.

Веб-сервис - это программный модуль, идентифицируемый с помощью URI (Unified Resource Identifier), внешние интерфейсы и связи которого определены и описаны с помощью XML. Другие программные системы должны иметь возможность находить описания сервисов. Эти программные системы могут взаимодействовать с веб-сервисом в стиле, определенном в его описании, используя XML-сообщения, передаваемые по некоторому интернет протоколу. Отсюда видно, что технология веб-сервисов базируется на

технологии XML, являющейся открытым стандартом. Таким образом, веб-сервисы позволяют приложениям взаимодействовать друг с другом независимо от платформы, на которой они развернуты, а также от языка программирования, на котором они написаны.

Существуют два вида архитектуры – базовая и расширенная, они отличаются набором операций. Базовая архитектура веб-сервисов предполагает наличие, по крайней мере, трех организаций: провайдера сервиса, регистра сервисов, обеспечивающего регистрацию и средства поиска сервисов, и клиента, запрашивающего сервис. Базовая архитектура (Basic Architecture) включает следующие технологические операции:

- обмен сообщениями
- описание веб-сервисов
- публикация и поиск описаний веб-сервисов.

Расширенная архитектура веб-сервисов использует технологии и компоненты, определенные в базовой архитектуре и включает в себя дополнительные функции и возможности. Веб-сервисы базируются на следующих основных веб-стандартах:

- XML (Расширяемый язык разметки, предназначенный для хранения и передачи структурированных данных)
- SOAP (Протокол обмена сообщениями на базе XML)
- WSDL (Язык описания веб-сервисов)
- UDDI (Универсальный интерфейс распознавания, описания и интеграции)

Язык XML играет важную роль в веб-сервисах. Он является основой для таких технологий, как SOAP и WSDL, а также определяет формат данных, используемый для обмена информацией между потребителем сервиса и самим сервисом.

Базовым протоколом, обеспечивающим взаимодействие в среде веб-сервисов, является протокол SOAP (Simple Object Access Protocol). Протокол SOAP — это основанный на языке XML стандарт для обмена сообщениями в распределенных средах (WWW). Он предназначен для удаленного вызова методов. SOAP можно использовать с разными транспортными протоколами, включая HTTP, SMTP.

Интерфейс для доступа к веб-сервису описывается на языке описания WSDL (Web Services Description Language) основанном на XML, и содержит всю информацию, необходимую для доступа к данному сервису. С точки зрения WSDL-документа веб-сервис представляет собой коллекцию портов, которые, в свою очередь, являются коллекцией абстрактных операций и сообщений. Абстракция операций и сообщений позволяет связывать их с различными протоколами и форматами данных типа SOAP, HTTP GET/POST или MIME.

UDDI - это каталог веб-сервисов, доступный через Интернет.

В рамках концепции семантического веб'а W3C ввел понятие «семантического веб-сервиса». Семантический веб-сервис строится на базе стандартного веб-сервиса, над которым строится семантическое описание.

Суть семантического описания сервиса состоит в том, что синтаксические элементы, определяемые в WSDL описании сервиса, привязываются к терминам глобального информационного ресурса, так называемой онтологии предметной области. Эта привязка сводится к определению входов/выходов сервиса в терминах онтологии, определению предусловий выполнения сервиса и эффектов от его выполнения. Одному семантическому описанию может соответствовать несколько синтаксических описаний, программный код для которых, в свою очередь, может быть реализован в нескольких вариантах, с использованием разных языков программирования и выполняться на разных платформах. Даже определение входов/выходов сервиса в терминах онтологии, при условии, что онтология принимается как потребителями, так и производителями сервиса, существенно упрощает и уточняет поиск сервисов. Более того, в перспективе можно говорить о вызове сервиса не только по адресу (URL), но и по его семантическому описанию.

Для разработки веб-сервисов созданы специальные среды. При использовании этих сред обычно предполагается, что процесс создания сервиса начинается с разработки программного кода, по которому автоматически генерируется интерфейс сервиса в виде WSDL-описания. Этот подход к разработке веб-сервисов очень популярен, но перспективность его вызывает сомнения. Более перспективен подход к созданию сервисов, который начинается с разработки и утверждения его интерфейса, который, в соответствии с рекомендацией W3C, представляется в виде WSDL документа. К такому подходу подталкивает и перспектива разработки веб-сервисов как

коммерческих продуктов. Поэтому в диссертации целесообразно исследовать оба этих подхода, причем с применением каждого из них провести разработку практического приложения.

Кроме «простых» сервисов, методы которых не используют обращения к другим сервисам, широкое применение находят и композитные сервисы, которые при исполнении могут обращаться, причем неоднократно, к другим сервисам. Разработка композитных сервисов может проводиться непосредственно на языках типа Java, C# или с использованием специальных языков, ориентированных на композицию.

Наряду с четко определенной функциональностью веб-сервис характеризуется нефункциональными характеристиками. Именно эти характеристики ответственны за то, что приложение, отправившее запрос к веб-сервису, находится в значительной степени в условиях неопределенности. Это, прежде всего, относится к композитным сервисам. Наличие нефункциональных характеристик ставит вопрос о моделировании процессов, происходящих при обращении к композитным сервисам, и вообще о моделировании приложений, разрабатываемых в архитектуре, ориентированной на веб-сервисы. Для этих целей очень удобно использовать раскрашенные сети Петри.

В настоящее время чаще всего используется статическая парадигма работы с веб-сервисами - приложение обращается к веб-сервису по конкретному адресу (URL), который был выбран при разработке приложения и вынуждено пассивно ждать ответа. Однако веб-сервисов с идентичной функциональностью может быть достаточно много, так как их разработка, поддержание и размещение, в конечном счете, определяется получением прибыли. Таким образом, если характеристики обслуживания конкретным веб-сервисом, расположенным на конкретном сервере, не удовлетворяют клиента, то желательно было бы обратиться к другому сервису на выбранном сервере или к тому же сервису, но расположенному на другом сервере.

На уровне кода приложения эта задача будет решена, если будет создана возможность обращаться к веб-сервису не только по конкретному URL, но и по некоторому семантическому коду, который представляет совокупность функционально идентичных веб-сервисов. В динамической парадигме работы с веб-сервисами эти функции выполняет сервисная шина предприятия. Именно она осуществляет мониторинг обращений к веб-сервисам и, опираясь на

информацию мониторинга, производит соединение по конкретному URL.

В диссертационном исследовании ставятся задачи анализа процессов разработки «простых» и композитных сервисов и задачи моделирования динамической парадигмы работы с веб-сервисами.

Во второй главе диссертации развивается подход к моделированию веб-сервисов и сервис-ориентированных приложений на базе многоцветных сетей Петри.

На рис. 1 представлена общая модель веб-сервиса в виде сети Петри-Маркова. Модель конкретного сервиса определяется заданием параметров, а именно вероятностей разрешения конфликтных ситуаций, параметров распределения времени выполнения операций. Эта модель позволяет получить оценку статистических параметров функционирования сервиса.

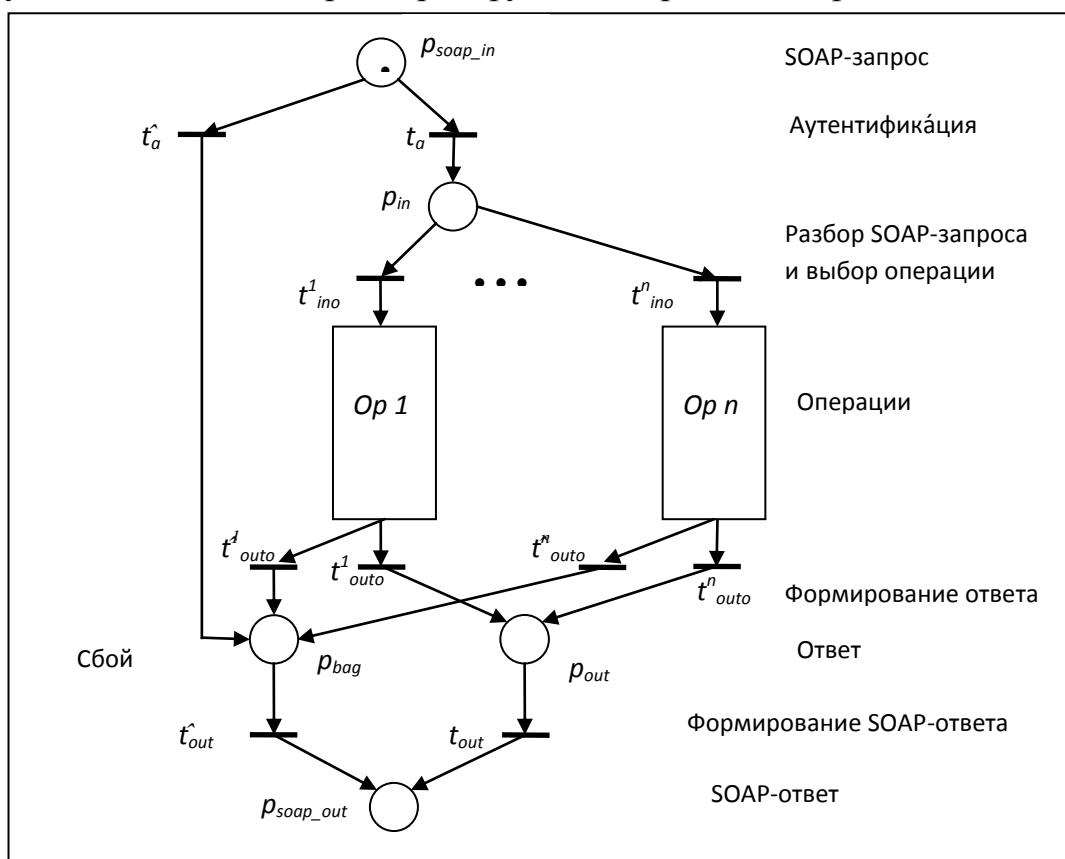


Рис. 1 Общая модель веб-сервиса в виде сети Петри-Маркова

Подсеть $\{p_{soap_in}; \hat{t}_a, t_a\}$ – моделирует конфликтную ситуацию при аутентификации. Требуется задать вероятность срабатывания перехода $\hat{t}_a - Ver(\hat{t}_a)$ и вероятность срабатывания перехода $t_a - Ver(t_a)$. При этом $Ver(\hat{t}_a) + Ver(t_a) = 1$.

Подсеть $\{p_{in}; t_{ino}^1, \dots, t_{ino}^n\}$ – моделирует ситуацию выбора операции. При каждом обращении к сервису выполняется только одна операция. Требуется задать вероятность срабатывания перехода $t_{ino}^i - Ver(t_{ino}^i)$, при этом $\sum Ver(t_{ino}^i) = 1$.

В языке семантического описания веб-сервиса OWL-S введено понятие атомарного процесса, соответствующего операции WSDL. В соответствии с OWL-S, модель процесса задается четверкой вида $\langle I, O, P, E \rangle$, I – вход процесса, O – выход процесса, P и E –предусловия и эффекты. P – предусловие выполнения процесса – логическое выражение, отражающее условия, накладываемые на состояние объектов предметной области, при соблюдении которых выполнение процесса будет успешным. E – эффект выполнения процесса – логическое выражение, связывающее входы процесса с изменением состояния объектов предметной области в результате успешного выполнения процесса. В соответствии с этим представлением об атомарном процессе модель операции, выполненная в виде сети Петри-Маркова, выглядит следующим образом (рис. 2):

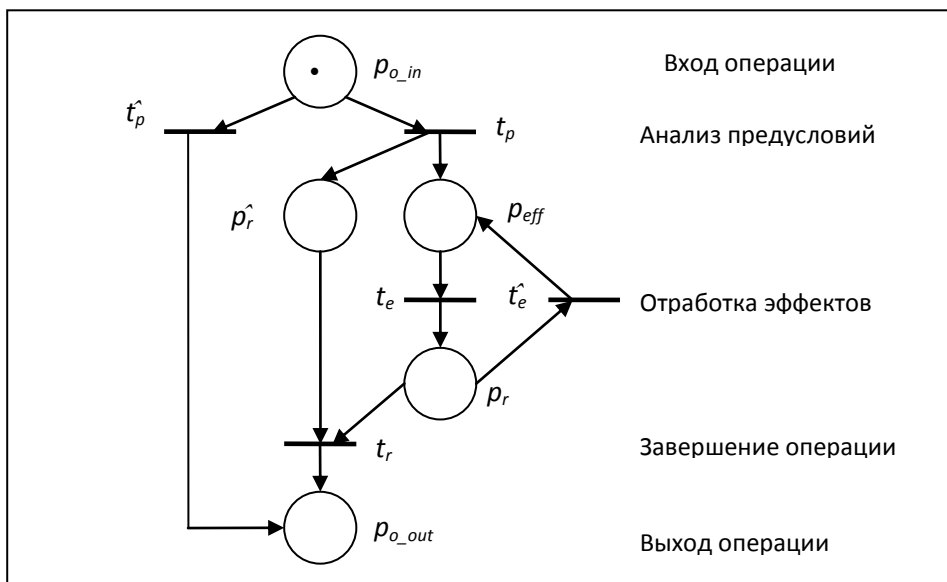


Рис. 2 Модель операции веб-сервиса в виде сети Петри-Маркова

Подсеть $\{p_{o_in}, \hat{t}_p, t_p\}$ – моделирует конфликтную ситуацию при анализе предусловий. Операция выполняется штатным образом, то есть получается правильный выход и корректные изменения в предметной области (например, правильное списание средств со счета), если предусловия операции выполнены. Требуется задать вероятность срабатывания перехода \hat{t}_p – $Ver(\hat{t}_p)$ и вероятность срабатывания перехода t_p – $Ver(t_p)$. При этом $Ver(\hat{t}_p) + Ver(t_p) = 1$.

Подсеть $\{p_r, \hat{t}_e, t_r\}$ – моделирует конфликтную ситуацию, возникающую при обработке эффектов. Переход \hat{t}_e срабатывает при неудачной попытке изменения внешней базы данных. При этом, $Ver(\hat{t}_e) + Ver(t_r) = 1$.

В модели операции подсеть, связанная с обработкой эффектов, в каждом конкретном случае требует более детальной разработки, если требуется отразить детали изменения объектов предметной области.

При проектировании бизнес приложений приходится использовать операции нескольких сервисов или нескольких операций одного сервиса. На рис. 3 показан клиент (это, в свою очередь, может быть веб-сервис), который последовательно вызывает операции разных (или одного) веб-сервисов, причем вход для операции второго сервиса формируется на базе выхода первого.

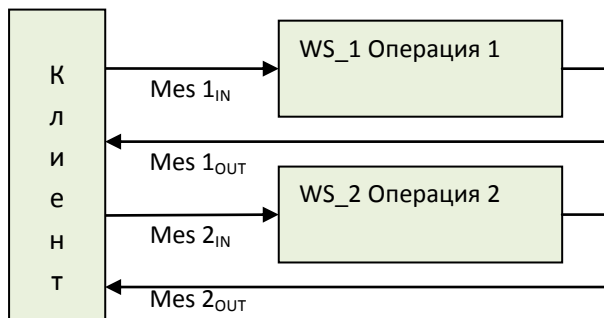


Рис. 3 Последовательный вызов операций веб-сервисов

Проблема заключается в том, что даже сервисы, имеющие на выходе и входе семантически идентичные сообщения, могут применять различную синтаксическую разметку. Например, выходное сообщение сервиса WS_1 представлено с использованием элементов XML, а вход сервиса WS_2 требует той же самой информации, представленной с использованием атрибутов XML. В данном случае приходится программировать преобразование. Чтобы упростить программирование клиента, можно определить композитный веб-сервис с операцией 1_2 с входным сообщением Mes_1 и выходным сообщением Mes_2 (рис. 4).

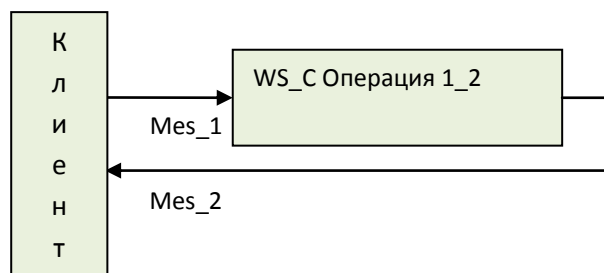


Рис. 4 Вызов композитного веб-сервиса

Создание модели композитного сервиса на базе сети Петри-Маркова предполагает, что все вопросы информационной совместимости сервисов разрешены. Графическая диаграмма, представляющая сеть композитного сервиса, позволяет более четко представить структуру композитного сервиса и оценить характеристики его выполнения. На рис. 5 представлена модель композитного сервиса, составленного из параллельно вызываемых операций других сервисов с обработкой получаемых результатов.

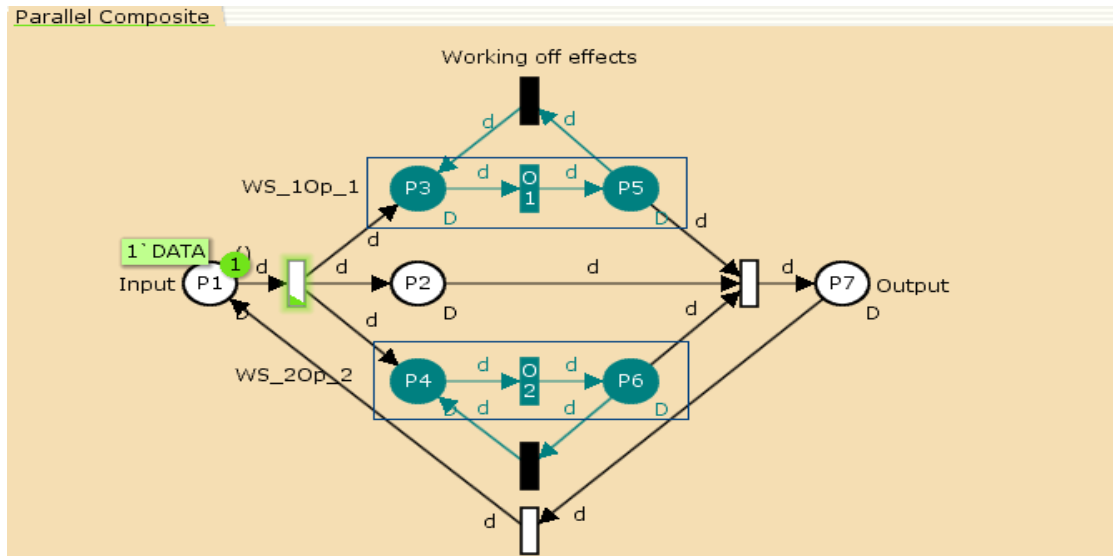


Рис. 5 Модель композитного сервиса, составленного из параллельно вызываемых операций двух сервисов. Зацикливание модели позволяет определить вероятность маркировки и граф достижимости.

P_i	1	2	3	4	5	6	7	Вероятность
M_0	1	0	0	0	0	0	0	0.19
M_1	0	1	1	1	0	0	0	0.21
M_2	0	1	0	1	1	0	0	0.12
M_3	0	1	0	0	1	1	0	0.19
M_4	0	0	0	0	0	0	1	0.18
M_5	0	1	1	0	0	1	0	0.11

Рис. 6 Вероятность маркировки

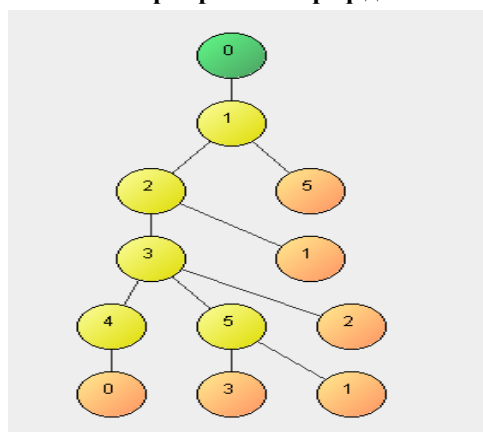


Рис. 7 Граф достижимости маркировки

Используя раскрашенные сети Петри можно промоделировать более тонкие особенности взаимодействия клиента и сервиса. Различают статическое и динамическое связывание клиента и веб-сервиса. При статическом связывании клиент обращается непосредственно к веб-сервису по конкретному адресу (URL), а при динамическом - по некоторому семантическому коду к некоторому посреднику - промежуточному компоненту, роль которого может выполнять сервисная шина предприятия (ESB). Раскрашенная сеть, моделирующая статическое связывание клиента и веб-сервиса, представлена на рис 8.

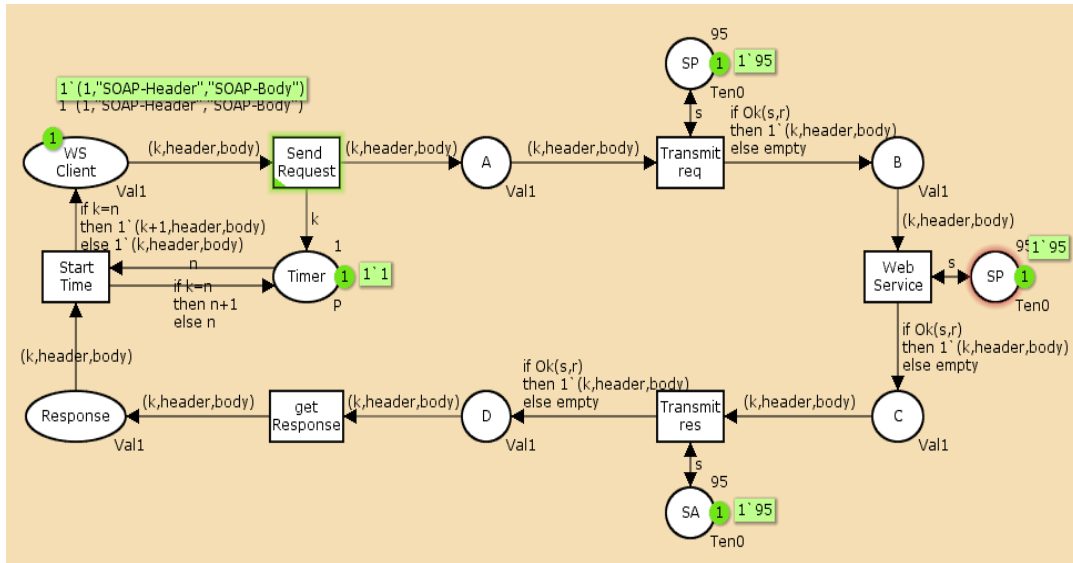


Рис. 8 Статическое связывание клиента и веб-сервиса

В диссертационном исследовании промоделировано два варианта подключения посредника к взаимодействию клиента и сервиса. В первом варианте все взаимодействие осуществляет посредник: клиент посылает ему запрос и получает ответ. На рис. 9 представлена сеть Петри, которая моделирует этот вариант динамического связывания клиента и веб-сервиса.

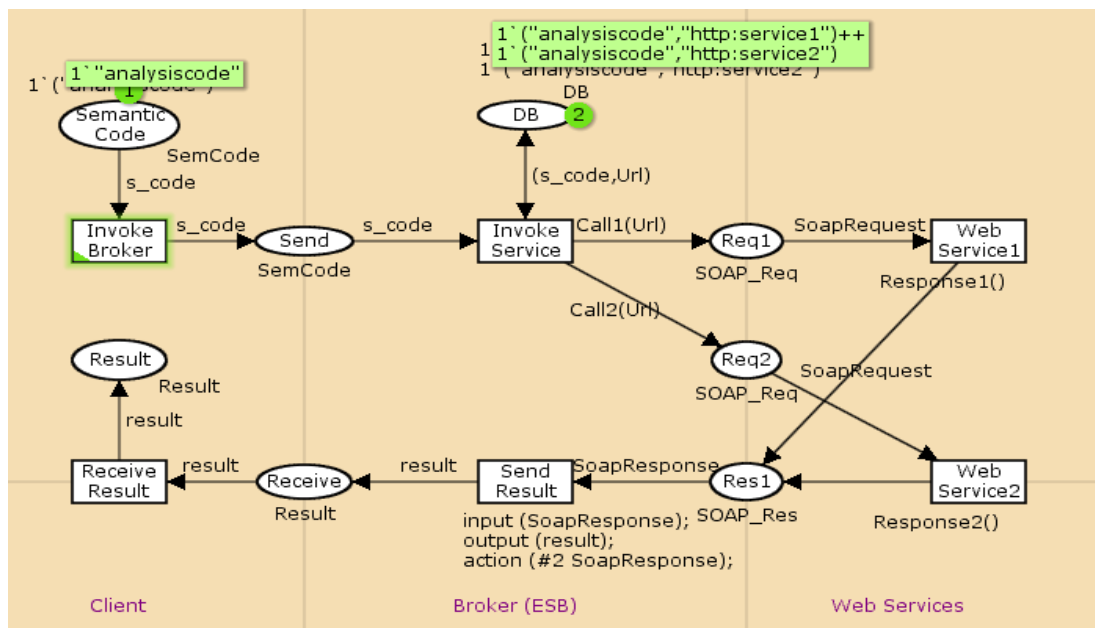


Рис. 9 Модель взаимодействия без непосредственного связывания с веб-сервисом

Во втором варианте в ответ на запрос посредник высылает клиенту URL веб-сервиса, который в текущей обстановке наиболее подходит для выполнения задач клиента. В этом варианте клиент берет на себя всю работу по

установлению связи с веб-сервисом. Модель этого варианта взаимодействия представлена на рис. 10.

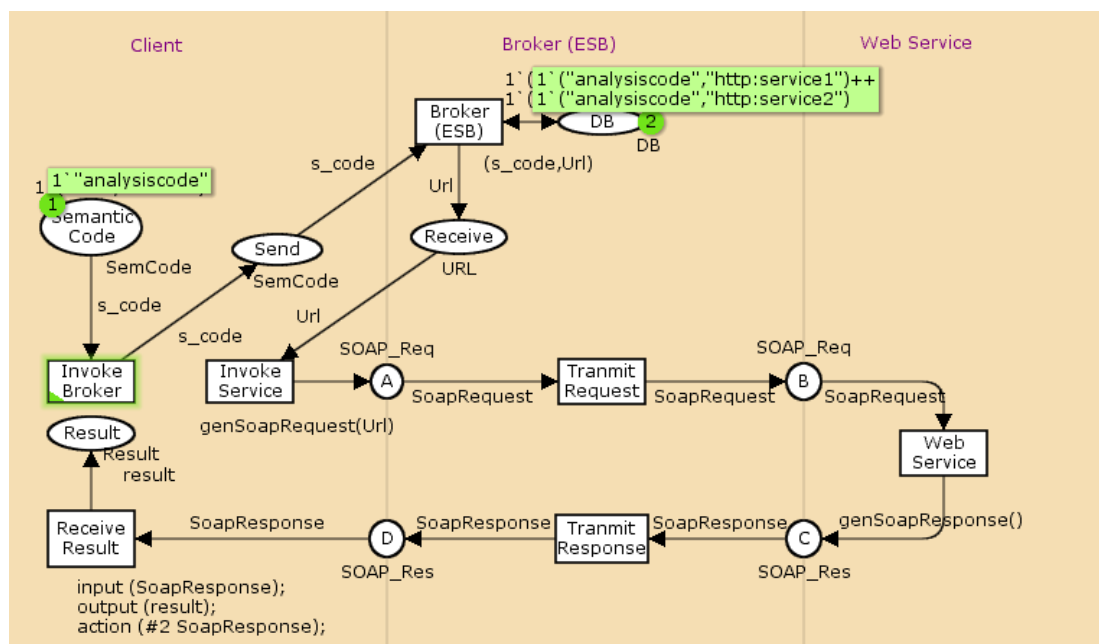


Рис. 10 Модель взаимодействия путем возвращения URL

Разработанный подход к моделированию веб-сервисов, а также их модели используются при создании модели мобильного приложения и веб-сервиса, взаимодействующих по каналу GPRS. Построение серии моделей на базе сетей Петри позволило, прежде всего, выяснить все детали, необходимые для разработки собственно мобильного приложения и веб-сервиса, а также оценить последствия потери сообщений, неизбежно происходящих при использовании беспроводных сетей передачи данных GPRS. Было построено несколько вариантов моделей на сетях Петри с помощью инструмента CPN Tools, далее представлены варианты, моделирующие работу реально разработанных мобильного приложения и веб-сервиса.

На рис. 11 показана иерархическая одноцветная модель взаимодействия мобильного приложения и веб-сервиса. Данная модель позволяет проследить основные потоки данных и события, происходящие в процессе взаимодействия мобильного приложения, построенного на базе КПК, и веб-сервиса, выполняющего различные операции с офисной базой данных. В реально разработанной системе каждый день в среднем обслуживается около 40 мобильных клиентов. Для более полного отражения в модели особенностей взаимодействия необходима многоцветная модель.

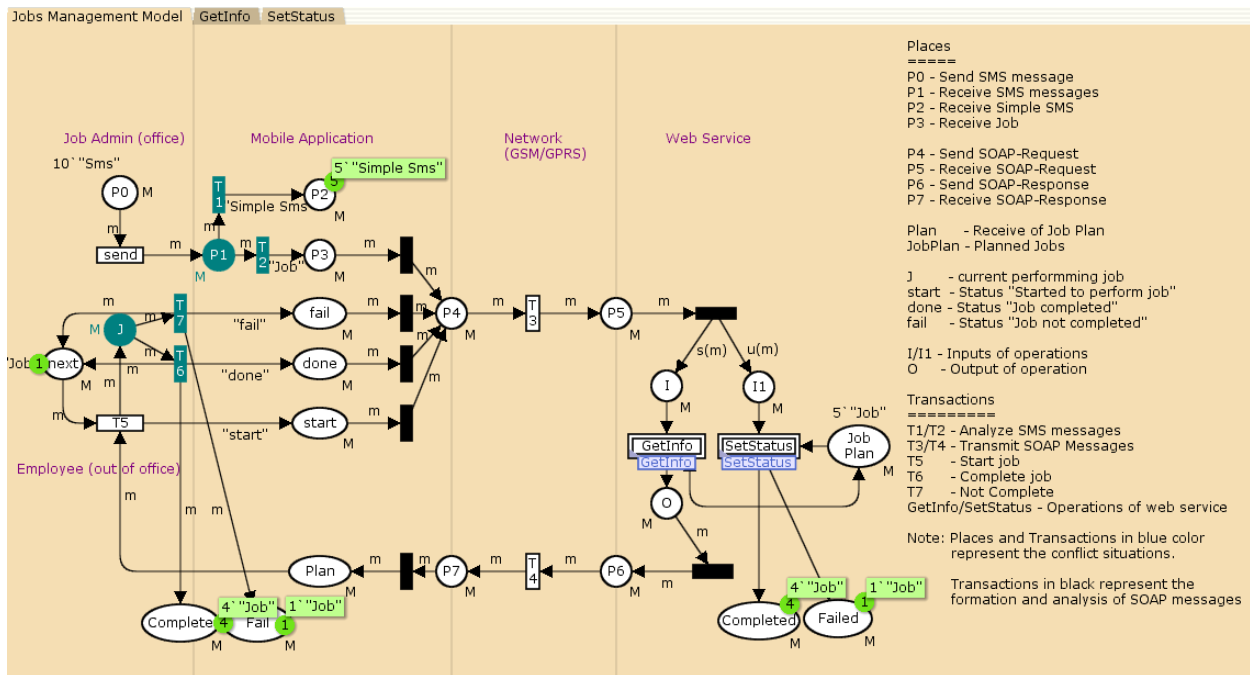


Рис. 11 Одноцветная модель взаимодействия мобильного приложения и веб-сервиса

На рис. 12 показано общее представление иерархической многоцветной модели взаимодействия мобильного приложения и веб-сервиса. В иерархических элементах модели (Employee, Mobile, GPRS, Web Service) детально рассматривается функциональность взаимодействующих компонентов моделируемой системы и передаваемые данные между ними.

Был разработан и отмоделирован простейший протокол взаимодействия мобильного приложения и веб-сервиса: если сообщение не было получено во время, отведенное для ожидания ответа, то посылается повторный запрос к веб-сервису. В реальной системе сообщения довольно часто теряются в канале GPRS. Анализировалась надежность взаимодействия при разном количестве повторных запросов и разных временах ожидания ответа. Для получения статистических результатов моделирования использован монитор, определяемый пользователем (User-defined monitors), который является более гибким по сравнению со стандартными мониторами CPN Tools, но требует дополнительного программирования. На рис.13 и 14 представлены примеры оформления результатов моделирования в виде столбиковых диаграмм.

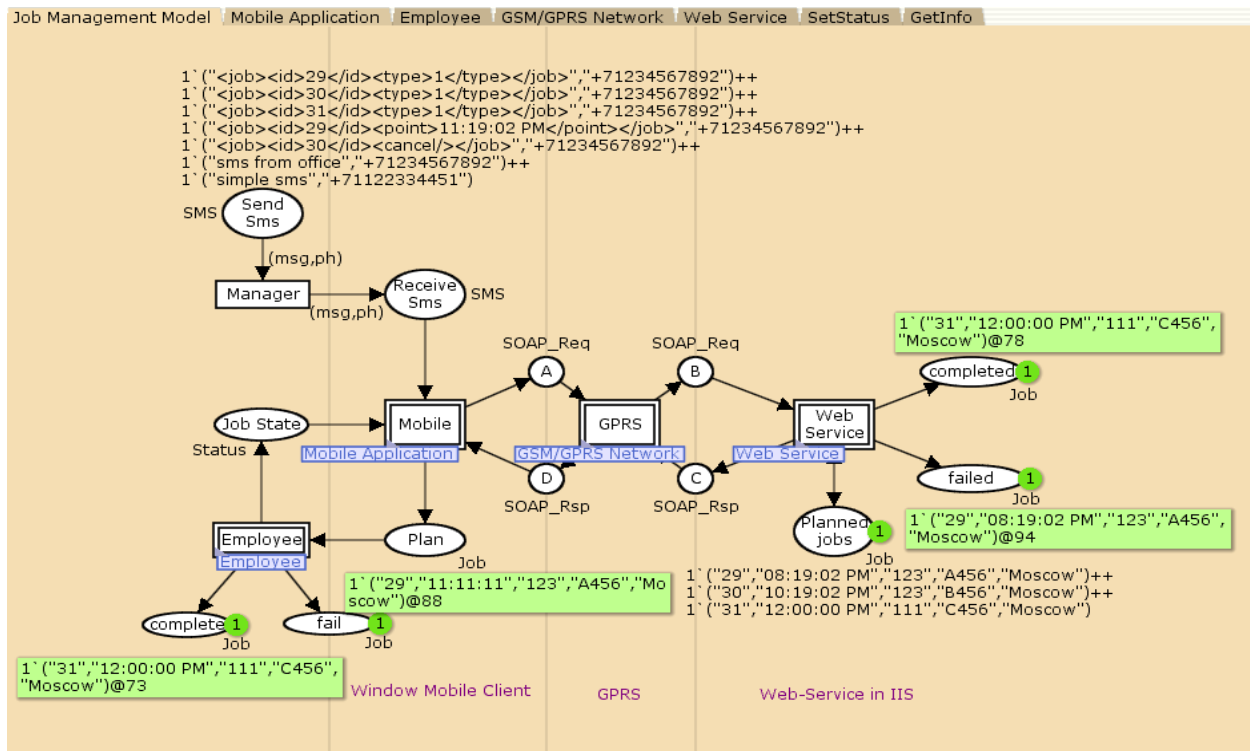


Рис. 12 Многоцветная модель взаимодействия мобильного приложения и веб-сервиса

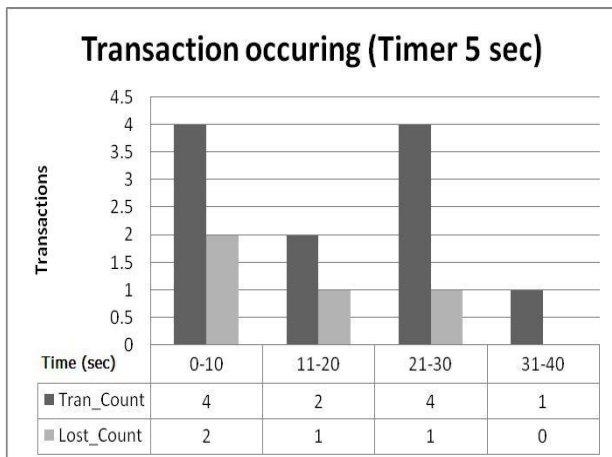


Рис. 13 Гистограмма количества транзакций и потери (таймер 5 сек)

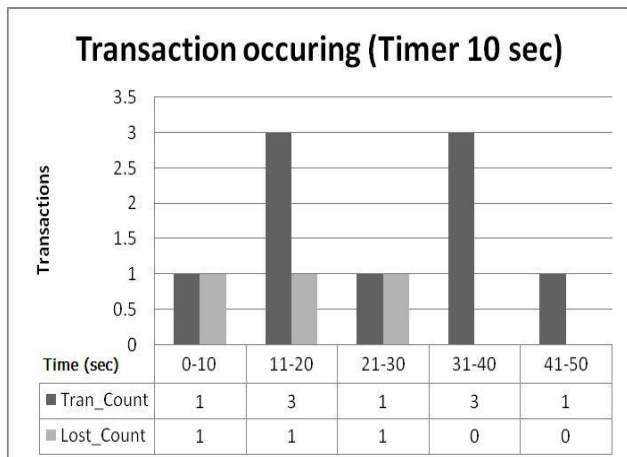


Рис. 14 Гистограмма количества транзакций и потери (таймер 10 сек)

В третьей главе диссертации проведен анализ технологических средств, используемых для разработки веб-сервисов и приложений, использующих веб-сервисы. Рассмотрены методы разработки атомарных и композитных веб-сервисов.

В работе над диссертацией анализировались две платформы для создания веб-сервисов. Платформа J2EE (Java 2 Enterprise Edition) позволяет создавать веб-сервисы на основе Java-приложений (Java класс, EJB компонент) для любых архитектур и операционных систем. Платформа Microsoft .NET

позволяет создавать веб-сервисы с помощью таких языков программирования как Visual Basic и C# на ASP.NET технологии. Анализ показывает, что J2EE является единственной платформой, действительно позволяющей создавать веб-сервисы для любых операционных систем, в то как Microsoft еще только создает интерпретатор CLR (Common Language Runtime) для выполнения своего промежуточного кода на операционных системах, отличных от Windows.

Подход к разработке веб-сервисов, названный в диссертации методом «снизу-вверх», предполагает первоначальное создание программного кода или использование существующего программного кода для превращения его в веб-сервис с последующей генерацией всех составляющих сервиса. В процессе создания веб-сервиса с использованием подхода «снизу-вверх» необходимо пройти следующие шаги:

- Создать приложение, реализующее функциональность веб-сервиса;
- Развернуть приложение на сервере и протестировать его;
- Сгенерировать WSDL описание веб-сервиса;
- Выполнить семантическое аннотирование веб-сервиса (необязательно);
- Создать клиентское приложение для доступа к веб-сервису.

Данный подход не конструктивен, когда разрабатываются крупные системы, так как без проектирования функциональности и взаимодействия веб-сервисов сразу начинать с уровня реализации отдельных веб-сервисов по меньшей мере нерационально.

Однако подход «снизу вверх» полезен для превращения существующего приложения в веб-сервис: создания веб-сервиса на основе бизнес-логики, содержащейся в таких существующих программных компонентах, как EJB или JavaBeans. При разработке веб-сервисов применение подхода «снизу вверх» позволяет разработчикам уменьшить потери времени на проектирование интерфейсов веб-сервисов, и облегчает их реализацию.

Подход к разработке веб-сервисов, названный в диссертации методом «сверху-вниз», предполагает первоначальное создание WSDL описания веб-сервиса и генерацию на его основании скелета программного кода и всех составляющих сервиса. В процессе создания веб-сервиса с использованием подхода «сверху-вниз» необходимо пройти следующие шаги:

- Определение WSDL описания веб-сервиса;
- Генерация скелета программного кода;
- Создание программного кода, реализующего функциональность веб-сервиса;
- Развертывание приложения на сервере;
- Семантическое аннотирование веб-сервиса (необязательно);
- Создание клиентского приложения для доступа к веб-сервису.

Используя подход «сверху-вниз», получается ясный, четко структурируемый программный код, который в дальнейшем не вызовет трудностей при сопровождении. Преимущества подхода «сверху-вниз» - это высокий уровень контроля, который разработчик имеет над интерфейсом сервисов и инкапсулируемой логикой. Данный подход устраняет проблемы функциональной совместимости компонентов при создании крупных систем в рамках сервис-ориентированной архитектуры.

В четвертой главе проведена экспериментальная проверка предлагаемых подходов к статической и динамической композиции веб-сервисов на основе разработанных приложений для моделирования процесса инкассации банкоматов и мобильного приложения по управлению заданиями персонала, выполняющего производственные задания вне офиса.

В экспериментальной работе выполнено две разработки: разработка системы прогнозирования инкассаций банкоматов и системы оперативного управления заданиями сотрудников. При разработке приложений рассматриваются оба подхода, предлагаемого в третьей главе.

Разработка системы моделирования процесса инкассации осуществляется по подходу «сверху-вниз». Разработанная система выполняется в виде композиции 4 веб-сервисов:

- GenerateRequestWS - веб-сервис для генерации потока запросов на выдачу наличности;
- PrognosisManagementWS - веб-сервис для формирования потока инкассаций и оценки эффективности работы банкомата при заданном потоке;
- PrognosisWS - веб-сервис отработки потока запросов текущей инкассации;

- OptimizeResultWS - веб-сервис определения оптимальной политики инкассации.

Реализация веб-сервисов для системы прогнозирования инкассации выполнена с использованием языка программирования Java. При реализации веб-сервисов в качестве средства разработки использовано средство NetBeans IDE, которое предоставляют возможность генерации «скелета» программного кода по существующему WSDL описанию. В данной разработке веб-сервис для формирования потока инкассаций является композитным. С целью тестирования функционального взаимодействия между программными компонентами, развернутыми на различных платформах, в данной разработке реализовано клиентское приложение на языке C# на платформе .NET с помощью программного средства Microsoft Visual Studio 2008.

Разработка системы оперативного управления заданиями сотрудников сервисного центра компании БПЦ происходила в условиях, когда база данных центра уже несколько лет находилась в условиях промышленной эксплуатации. Задача заключалась в том, чтобы создать мобильное приложение, взаимодействующее с этой базой через специальный веб-сервис. Сотрудники центра, работающие в «полевых» условиях, должны взаимодействовать с этой базой через мобильное приложение по каналам GPRS, используя этот веб-сервис. Так как информационное окружение, в котором должен функционировать сервис, четко определено, то при разработке веб-сервиса применялся подход «снизу вверх». Основными компонентами данной системы являются веб-сервис, выполняющий различные операции с офисной базой данных, и мобильное приложение. Основные программные компоненты были реализованы на базе технологии .NET с использованием средства Microsoft Visual Studio 2008. Веб-сервис разработан на языке C# на платформе .NET Framework 3.5, а программные модули мобильного приложения - на платформе .NET Compact Framework 3.5.

Разработанное приложение предполагает интенсивный обмен сообщениями между офисом и сотрудниками, позволяющий оценить состояние исполнения каждого задания и выдать корректирующие распоряжения по задачам. Очевидно, вопросы безопасности, хотя бы на элементарном уровне, должны быть решены. Следование стандартным подходам к обеспечению безопасности оказалось слишком сложным, поэтому для безопасного доступа к веб-сервису была реализована собственная процедура аутентификации. В

листинге. 1 показан формат SOAP-запроса, отправляемого мобильным клиентом к веб-сервису при вызове веб-метода. В заголовке SOAP сообщения присутствуют учетные данные пользователя для доступа к веб-сервису (в данном примере номер мобильного телефона).

Листинг. 1 Формат SOAP-запроса

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <authenticationHeader xmlns="http://sc.bpc.ru">
      <phoneNumber>string</phoneNumber>
    </authenticationHeader>
  </soap:Header>
  <soap:Body>
    <getInfo xmlns="http://sc.bpc.ru">
      <jobID>int</jobID>
    </getInfo>
  </soap:Body>
</soap:Envelope>
```

При разработке веб-сервиса был реализован метод, который считывает идентификационную информацию запрашивающей стороны из этого SOAP-запроса и проверяет, разрешен ли этому пользователю вход в веб-сервис. Особенностью данного решения является то, что этот метод не является веб-методом (то есть он не декларирован в качестве операции WSDL) и, прежде всего, выполняет функцию аутентификации при вызове любого веб-метода данного веб-сервиса. Если запрашивающая сторона не зарегистрирована в базе офиса, то вызываемый веб-метод возвращает SOAP-ответ, содержащий сообщение о том, что доступ к веб-сервису запрещен, а в протокол нарушения правил доступа заносится запись о факте несанкционированного доступа. Пример данного сообщения представлен в листинге. 2, а формат записи о факте несанкционированного доступа – в листинге. 3.

Листинг. 2 Пример сообщения, которое содержится в SOAP-ответе

```
<error>
  <message>Invalid user access to this Web service</message>
  <source>MobileWebService</source>
</error>
```

Листинг. 3 Формат записи о факте несанкционированного доступа

```
<unauthorizedAccess>
  <date>yyyymmdd</date>
  <time>hhmmss</time>
  <phoneNumber>string</phoneNumber>
</unauthorizedAccess>
```

К этому следует добавить, что все допустимые обращения к веб-сервису также протоколируются. Предлагаемое решение не сложно в реализации, но, тем не менее, защищает систему от нежелательных обращений, так как недооценка возможных угроз, имеющих при работе в открытом пространстве сети Интернет, может привести к нежелательным последствиям.

Отладка разработанного мобильного приложения проводилась на одном из существующих эмуляторов в комплекте средства разработки Windows Mobile 6 Software Development Kit, называемом Windows Mobile 6 Professional Emulator.

Метод «сверху-вниз», использованный для разработки веб-сервисов в системе моделирования процесса инкассации банкоматов, доказал свою эффективность. В результате тщательного продумывания элементов интерфейса и фиксации их в виде WSDL описания разработка программного кода существенно упростилась.

Разработка и отладка взаимодействия веб-сервиса и мобильного клиента успешно завершились в значительной степени благодаря моделированию на сетях Петри и использованию эмулятора мобильного устройства.

В заключении приведены основные результаты диссертационной работы.

В приложении вынесен акт о внедрении, а также исходные коды, разработанные в ходе данной диссертационной работы.

ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

1. Исследована методика разработки веб-сервисов, как по методу «снизу-вверх», при котором разработка начинается с создания программы, с последующей автоматической генерацией интерфейсного окружения, так и по методу «сверху-вниз», при котором разработка начинается с создания интерфейса с последующей автоматической генерацией «скелета» программы.

2. Рассмотрены вопросы построения композитных сервисов.

3. Разработаны модели статической и динамической композиции веб-сервисов.

4. Разработаны модели взаимодействия мобильных приложений и веб-сервиса.

5. Разработано приложение, выполненное на базе композиции веб-сервисов, на платформе J2EE по методу «сверху-вниз».

6. Разработано приложение, выполненное в виде веб-сервиса и мобильных клиентов, для сервисного центра компании БПЦ и внедрено в промышленную эксплуатацию для оперативного управления действиями сотрудников, работающих в «полевых» условиях.

ОСНОВНЫЕ ПУБЛИКАЦИИ ПО ТЕМЕ ДИССЕРТАЦИИ

Публикации в журналах, входящих в перечень ВАК:

1. Аунг Аунг Хейн, Щукин Б.А., «Моделирование сервис-ориентированного приложения с помощью раскрашенных сетей Петри», Естественные и технические науки, № 6, 2011 г. Стр. 527-529.

2. Аунг Аунг Хейн, Щукин Б.А., «Моделирование сервис-ориентированного приложения на эмуляторе», Естественные и технические науки, № 6, 2011 г. Стр. 523-526.

3. Аунг Аунг Хейн, Щукин Б. А., «Обеспечение информационной безопасности при взаимодействии с веб-сервисом», Безопасность информационных технологий, №1, 2012 г., С. 124-127.

Работы, опубликованные в сборниках трудов конференций:

4. Аунг Аунг Хейн, Климов В.В., «Задача связи сервис-ориентированного приложения и мобильного устройства». В сборнике докладов «VIII Курчатовская молодежная научная школа», Москва, 2011 г. Часть 3. С. 166-170.

5. Аунг Аунг Хейн, Щукин Б.А., «Веб-сервис для моделирования процесса инкассации», Труды XIX международного научно-технического семинара «Современные технологии в задачах управления, автоматизации и обработки информации». Алушта, 2010, стр. 251-252.

6. Аунг Аунг Хейн, «Разработка взаимодействия мобильного устройства и веб-сервиса», XIV международная телекоммуникационная конференция студентов и молодых ученых «Молодежь и наука». Научная сессия НИЯУ МИФИ-2011.

7. Аунг Аунг Хейн, Щукин Б.А., «Моделирование взаимодействия мобильного приложения и веб-сервиса», Труды XX международного научно-технического семинара «Современные технологии в задачах управления, автоматизации и обработки информации». Алушта, 2011, стр. 37-38.