

Учреждение образования
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»

УДК 004.05

НЕБОРСКИЙ
Сергей Николаевич

**МОДЕЛИ, МЕТОДЫ И СРЕДСТВА РАЗРАБОТКИ И ОЦЕНКИ
КАЧЕСТВА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В УСЛОВИЯХ
ИЗМЕНЯЮЩИХСЯ ТРЕБОВАНИЙ**

АВТОРЕФЕРАТ
диссертации на соискание ученой степени
кандидата технических наук

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

Минск 2010

Работа выполнена в учреждении образования «Белорусский государственный университет информатики и радиоэлектроники»

Научный руководитель:

Бахтизин Вячеслав Вениаминович,
кандидат технических наук, доцент,
заведующий кафедрой программного
обеспечения информационных
технологий учреждения образования
«Белорусский государственный
университет информатики и
радиоэлектроники»

Официальные оппоненты:

Голенков Владимир Васильевич,
доктор технических наук, профессор,
заведующий кафедрой интеллектуальных
информационных технологий учреждения
образования «Белорусский
государственный университет
информатики и радиоэлектроники»

Железко Борис Александрович,
кандидат технических наук, заведующий
кафедрой экономической информатики
учреждения образования «Белорусский
государственный экономический
университет»

Оппонирующая организация:

**Белорусский государственный
университет**

КРАТКОЕ ВВЕДЕНИЕ

Создание качественного программного средства (ПС) – сложный процесс, который требует выполнения большого числа самых разных работ и решения различного рода задач. Разработка качественного ПС предполагает следование определенным стандартам и методам, использование эффективных инструментов и технологий, управление персоналом и взаимодействие с заказчиком. *Проблема обеспечения качества ПС* является важной и актуальной, о чем свидетельствуют принятые в Республике Беларусь стандарты (ГОСТ 28195-99, СТБ ИСО/МЭК 9126-2003, СТБ ISO/IEC 25000-2009). Данные стандарты определяют качество ПС и способствуют его оценке, саму же проблему обеспечения качества они не решают.

Об *актуальности проблемы обеспечения качества ПС* говорит и то, какое внимание ей уделяется Парком Высоких Технологий (Государственное учреждение «Администрация Парка высоких технологий», ПВТ) Республики Беларусь. Например, постоянно проводятся конференции с участием зарубежных специалистов (SQA Days 2008, SEF-2009, SEF-2010). ПВТ тесно сотрудничает с Международным институтом качества программного обеспечения (Германия), что также доказывает важность проблемы. Исследованиями в области качества ПС активно занимается и негосударственная научно-технологическая ассоциация «Инфопарк».

Обеспечение качества ПС имеет свои особенности при разработке *в условиях изменяющихся требований*. По данным исследований Центра Прикладного Программного Обеспечения университета Бозена (Италия) в 75 % современных программных проектов требования к ПС изменяются за время разработки, причем в 19 % из их числа изменяется более половины требований. При разработке ПС в условиях изменяющихся требований, как правило, применяются *гибкие методы*. По данным исследований на 2010 г., 43 % мировых компаний ведут разработку на их основе. Однако данные методы еще не предлагают эффективного решения проблемы обеспечения качества ПС в тех случаях, когда требования к ПС подвержены частым изменениям, т.к. предполагают фиксацию требований в начале очередного инкремента разработки. Что касается оценки качества ПС, то существующие модели качества недостаточно полно учитывают особенности разработки на основе гибких методов, и, следовательно, проблема оценки качества ПС остается нерешенной.

Диссертационная работа содержит комплексное решение для разработки ПС в условиях часто изменяющихся требований. Предложена эволюционно-инкрементная модель жизненного цикла (ЖЦ) ПС для гибких методов, которая определяет процессы, работы и задачи ЖЦ, их последовательность и взаимосвязь. В дополнение к ней предложена модель распределения задач между участниками проекта. Предложен метод организации процесса разработки ПС в условиях часто изменяющихся требований, который позволяет повысить качество ПС. Разработаны инструментальные средства для поддержки данного метода. Предложена модель качества ПС, позволяющая

более полно учитывать особенности разработки на основе гибких методов при оценке качества, а также модификация метода оценки качества.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Связь работы с крупными научными программами, темами

Исследования, положенные автором в основу диссертационной работы, а также выполняемые им в ходе работы, нашли отражение в следующих НИР:

– Грант Министерства Образования Республики Беларусь «Разработка методов и программных средств обеспечения качества программных продуктов» (№ ГР 20090676, ГБЦ № 09-3104, период выполнения – с 01.03.2009 по 31.12.2009).

– «Разработать модели, методы, алгоритмы для оценки качества и состояния, обеспечения отказоустойчивости, защищенности, диагностируемости аппаратно-программных средств сложных систем и внедрения в современные обучающие комплексы» (ГБ № 06-2004, период выполнения – с 02.01.2006 по 31.12.2010).

Цель и задачи исследования

Целью диссертационной работы является создание моделей и методов разработки и оценки качества ПС, учитывающих специфику часто изменяющихся требований.

Под *качеством ПС* в данной работе понимается весь объем признаков и характеристик ПС, который относится к его способности удовлетворять установленным или предполагаемым потребностям.

Обеспечение качества ПС – все запланированные и систематически выполняемые в рамках системы качества работы, при необходимости, объективные доказательства, обеспечивающие уверенность в том, что объект будет полностью соответствовать установленным требованиям качества.

Цель определяет следующие *задачи* исследования:

- анализ моделей ЖЦ ПС, методов разработки ПС, моделей качества ПС;
- разработка модели ЖЦ ПС и модели распределения задач ЖЦ ПС между участниками проекта для гибких методов;
- создание метода организации процесса разработки ПС в условиях часто изменяющихся требований и инструментальных средств его поддержки;
- разработка ориентированной на использование гибких методов модели качества ПС;
- экспериментальные исследования полученных результатов.

Объектом исследования являются процесс разработки и оценка качества ПС в условиях изменяющихся требований.

Предметом исследования являются гибкие методы разработки ПС.

Предполагается, что разработка новых моделей, методов и инструментальных средств позволит создавать более качественные ПС при разработке на основе гибких методов в условиях часто изменяющихся

требований и учитывать специфику разработки в таких условиях при оценке качества ПС.

Положения, выносимые на защиту

1. *Эволюционно-инкрементная модель ЖЦ ПС* для разработки на основе гибких методов, которая определяет процессы, работы, задачи и их связи и выделяет два вида итераций в процессе разработки: внешние (реализуют эволюционную стратегию разработки) и внутренние (реализуют инкрементную стратегию разработки). Преимуществами данной модели перед аналогами являются: возможность модификации требований к ПС в любой момент времени внутри внешней итерации за счет связи между работами по анализу требований, возможность быстрее получать готовые для использования промежуточные версии ПС за счет разделения на внутренние и внешние итерации, повышение качества ПС в первую очередь за счет учета функциональных требований пользователя фактически в режиме реального времени. Модель позволяет повысить сопровождаемость ПС более чем на 20 %.

2. *Метод организации процесса разработки ПС в условиях часто изменяющихся требований*, который базируется на применении эволюционно-инкрементной модели ЖЦ ПС и соответствующей ей модели распределения задач между участниками проекта, а также предлагает набор принципов разработки ПС (прямое создание архитектуры ПС на основе требований, управление реализацией изменяющихся требований, ориентация на быструю модификацию ПС на этапе сопровождения) и их конкретную реализацию. Его преимуществами перед аналогами являются устойчивость процесса разработки к изменению требований в произвольный момент времени и возможность быстрой модификации ПС после сдачи в эксплуатацию, что позволяет повысить качество ПС более чем на 15 %.

3. *Модель качества ПС при разработке на основе гибких методов*, которая определяет характеристики и подхарактеристики качества, адаптирует существующие и предлагает новые внешние метрики качества. Новыми метриками качества являются: для оценки подхарактеристик функциональности – приобретенная функциональность, точность вычислений, зависимость от внешних подсистем; надежности – покрытие тестами, степень автоматического восстановления; эффективности – соотношение времени отклика и затрат памяти, распределение длительных операций, сопровождаемости – повторное использование решений, сложность конфигурирования, способность к открытию интерфейса прикладного программирования, время обновления у пользователя, успешность изменений. Данная модель является основой для модификации метода оценки качества ПС с целью более полного учета особенностей разработки при применении гибких методов (полнота учета этих особенностей обеспечивается новыми предложенными метриками качества, число которых составляют более 30 % от общего числа входящих в модель метрик).

Личный вклад соискателя

Все вошедшие в диссертационную работу результаты были получены лично автором. Научный руководитель принимал участие в постановке задач, определении возможных путей решения, оценке результатов. В публикациях с соавторами вклад соискателя определяется рамками излагаемых в диссертационной работе результатов.

Апробация результатов диссертации

Разработки, положенные в основу диссертационной работы, и результаты исследований докладывались и обсуждались на V и VI международных научных конференциях «Информационные системы и технологии» (Минск, 2009, 2010), Международной конференции «Обеспечение качества программных средств SQA Days» (Минск, 2008), VI международной научно-практической конференции УкрПРОГ'2008 (Киев, 2008), X и XI республиканских научных конференциях студентов и аспирантов «Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях» (Гомель, 2007, 2008), Международной научно-практической конференции «Инженерно-педагогическое образование: проблемы и пути развития» (Минск, 2004), IV, V, VI международных научно-методических конференциях «Дистанционное обучение – образовательная среда XXI века» (Минск, 2004, 2005, 2006), 41-й и 42-й научных конференциях аспирантов, магистрантов и студентов БГУИР (Минск, 2005, 2006), Международных научно-практических конференциях «Современная радиоэлектроника: научные исследования и подготовка кадров» (Минск, 2006, 2007, 2008), Международной научно-практической конференции «Современные информационные компьютерные технологии mCT-2008» (Гродно, 2008), IV международной научно-практической конференции «Образование и наука без границ» (Прага, 2008).

Опубликованность результатов диссертации

По материалам выполненных исследований опубликовано 23 научных работы, в том числе 10 статей в рецензируемых изданиях (из них 7 соответствуют требованиям к объему научных статей на соискание степени кандидата наук), 13 докладов на научных конференциях. Без соавторства опубликовано 5 работ, из них 1 статья в рецензируемом издании. Общий объем публикаций составляет 6,2 авторского листа.

Структура и объем диссертации

Основное содержание работы изложено на 112 страницах машинописного текста, иллюстрировано 23 рисунками на 9 страницах, содержит 19 таблиц на 14 страницах. Диссертация состоит из введения, общей характеристики работы, четырех глав, заключения, библиографического списка, включающего 87 позиций, и четырех приложений на 37 страницах. Общий объем работы – 149 страниц.

В первой главе представлен анализ моделей ЖЦ ПС, методов разработки ПС, моделей качества ПС. Во второй главе предложены модели и метод, направленные на обеспечение более высокого качества ПС. В частности, предложены эволюционно-инкрементная модель ЖЦ ПС, модель распределения задач, метод организации процесса разработки ПС в условиях часто изменяющихся требований. В третьей главе предложены модель качества ПС при разработке на основе гибких методов и модификация метода оценки качества. В четвертой главе представлены результаты практических исследований, доказывающие преимущества предложенных моделей и метода организации процесса разработки, а также приведены описания реализованных инструментальных средств программной поддержки предложенного метода.

ОСНОВНОЕ СОДЕРЖАНИЕ ДИССЕРТАЦИОННОЙ РАБОТЫ

Во введении определена актуальность проблемы обеспечения и точной оценки качества ПС, показано, что при разработке ПС на основе гибких методов в условиях часто изменяющихся требований данная проблема остается нерешенной.

В общей характеристике работы сформулированы цели и задачи исследования, показана научная и практическая значимость полученных результатов исследования, изложены основные положения, выносимые на защиту.

В первой главе проведен анализ моделей ЖЦ ПС, методов разработки ПС, моделей качества ПС при их применении в условиях часто изменяющихся требований. Показано, что модели ЖЦ ПС, реализующие основные стратегии разработки, не позволяют создавать качественные ПС в указанных условиях из-за определения требований в начале ЖЦ (каскадные и инкрементные модели) или длины итерации и сложной структуры модели (эволюционные модели).

Проанализированы гибкие методы разработки (FDD, XP, Scrum, DSDM, ASD и др.), выявлены их недостатки при использовании в условиях часто изменяющихся требований: отсутствие возможности модифицировать требования к ПС внутри инкремента разработки, неуправляемость внесения изменений в ПС, отсутствие механизмов быстрой модификации ПС на этапе сопровождения, акцент на управлении проектом или тестировании ПС, но не на непосредственно реализации требований.

Рассмотрены основные модели качества ПС (модели МакКола, Бозма, FURPS/FURPS+, Дроми), а также модели качества по международным стандартам ISO/IEC 9216-1:2001 (в Республике Беларусь ему соответствует СТБ ИСО/МЭК 9126:2003) и ISO/IEC 2501n. Выявлены их недостатки: использование характеристик и метрик качества, которые не отвечают современным тенденциям разработки ПС; выстраивание неверной иерархии характеристик и подхарактеристик качества; отсутствие метрик, позволяющих в полной мере учитывать особенности разработки на основе гибких методов.

Исходя из проведенного анализа моделей и методов, сформулированы задачи исследования.

При исследовании текущего состояния предметной области особое внимание уделено работам таких российских ученых, как В. Благодатских (Московский государственный университет экономики, статистики и информатики), В. Бураков (Санкт-Петербургский государственный университет аэрокосмического приборостроения), В. Волнин, К. Посакалов, А. Воронцов, Д. Полищук, а также таких зарубежных исследователей, как К. Бек (Университет Орегона, США), У. Каннингем (Университет Пурдю, Индиана, США), Б. Бозм (Университет Южной Калифорнии, США), Дж. Де Люка, Р. Джеффрисон, К. Швабер, Р. Стефенс.

Во второй главе предложена эволюционно-инкрементная модель ЖЦ ПС для разработки на основе гибких методов. На рисунке 1 данная модель представлена в виде графа $G(V, E)$, вершины V которого соответствуют работам и процессам, а дуги E – их связям.

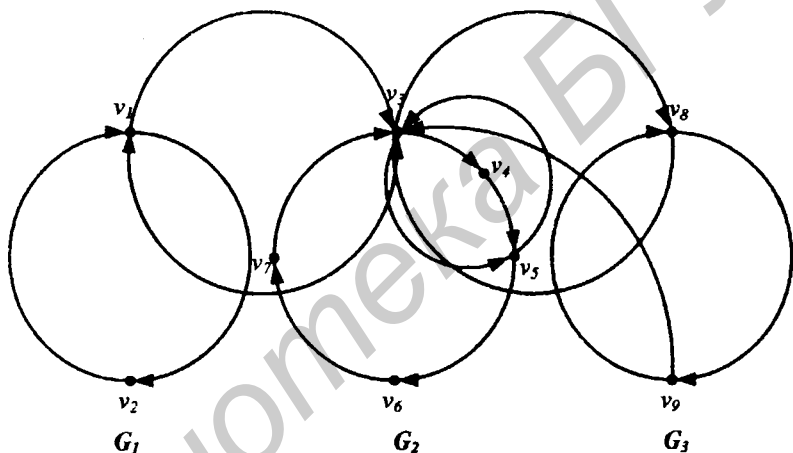


Рисунок 1 - Эволюционно-инкрементная модель ЖЦ ПС

Процесс заказа состоит из работ, предназначенных для определения проблемы, стоящей перед заказчиком, и пути ее решения. Данный процесс описывается подграфом $G_1(V_1, E_1)$, где $V_1 = \{v_1, v_2\}$: v_1 – анализ высокоуровневых требований заказчика к программным объектам системы; v_2 – планирование проекта и оценка требований.

Процесс разработки состоит из работ, предназначенных для непосредственно реализации требований к ПС и их проверки. Данный процесс описывается подграфом $G_2(V_2, E_2)$, где $V_2 = \{v_3, v_4, v_5, v_6, v_7\}$: v_3 – анализ требований к компонентам ПС; v_4 – проектирование программной архитектуры; v_5 – программирование и тестирование ПС; v_6 – сборка ПС; v_7 – квалификационные испытания ПС.

Отличительной чертой данной модели ЖЦ ПС является то, что процесс разработки предлагается строить итеративно с использованием двух видов

итераций: внешних и внутренних. Внутренней итерации соответствует множество вершин $\{v_3, v_4, v_5\}$ и множество дуг $\{(v_3, v_4), (v_4, v_5), (v_5, v_3), (v_3, v_3)\}$. Особенностью внутренней итерации является то, что она включает лишь ограниченный набор работ процесса разработки и допускает параллельное проектирование и программирование. Цель внутренних итераций – инкрементное наращивание функциональности ПС.

Внешней итерации соответствует множество вершин $\{v_3, v_4, v_5, v_6, v_7\}$ и множество дуг $\{(v_3, v_4), (v_4, v_5), (v_5, v_6), (v_6, v_7), (v_7, v_3)\}$. Цель внешней итерации – реализация прошедшей квалификационные испытания версии ПС, в которой реализован набор требований. Данный набор требований рассматривается как целое, и как целое представляет интерес для заказчика.

Рекомендуется ограничивать итерации по времени. для внутренних итераций следует использовать одно- или двухдневный временной блок, для внешних – одно- или двухнедельный.

Процесс эксплуатации определяет работы и задачи оператора. *Процесс сопровождения* заключается в поддержке разработанного ПС после сдачи в эксплуатацию. Предложенная модель ЖЦ не предполагает концентрации усилий на данных процессах, они не детализируются. Им соответствует подграф $G_3(V_3, E_3)$, где $V_3 = \{v_8, v_9\}$: v_8 – процесс эксплуатации; v_9 – процесс сопровождения.

Данная модель ЖЦ ПС сочетает эволюционную и инкрементную стратегии разработки. Наиболее близкими ее аналогами являются структурная эволюционная модель быстрого прототипирования и модель ЖЦ экстремального программирования. Преимуществами предложенной модели перед ними являются:

1) возможность модификации требований к ПС в любой момент времени внутри внешней итерации за счет связи между работами по анализу требований;

2) возможность быстрее получать готовые для использования промежуточные версии ПС за счет разделения на внутренние и внешние итерации, а также за счет отказа от детального проектирования ПС;

3) повышение качества ПС в первую очередь за счет учета функциональных требований пользователя фактически в режиме реального времени (в результате экспериментов установлено повышение сопровождаемости ПС более чем на 20 %);

4) преимущества, получаемые за счет сочетания эволюционной и инкрементной стратегий разработки:

- возможность получения функционального продукта после каждого инкремента (каждой внутренней итерации);

- снижение рисков по сравнению с каскадной моделью;

- отсутствие необходимости в полном определении функциональности ПС в начале ЖЦ.

Во второй главе в дополнение к эволюционно-инкрементной модели ЖЦ ПС предложена модель распределения задач ЖЦ между участниками проекта.

Предложенная модель описывается ориентированным графом $G(X, Y)$, где $X = \{x_i, i = \overline{1, n_x}\}$, $|X| = n_x$ – множество участников проекта, $Y = \{y_j, j = \overline{1, n_y}\}$, $|Y| = n_y$ – множество связей между ними, устанавливаемых на основе задач ЖЦ.

Проблема выбора распределения задач, обеспечивающего эффективную работу команды, сводится к *разбиению графа* $G(X, Y)$ на части $G_k = (X_k, Y_k)$, $X_k \subseteq X$, $Y_k \subseteq Y$, $\bigcup_{k \in K} G_k = G$, причем $\forall G_k, G_l$ таких, что $G_k \neq G_l$, выполняется $X_k \cap X_l = 0$. Для сравнения различных вариантов распределения задач предложено использовать коэффициент разбиения графа:

$$K_G = \sum_{k=1}^{N_k} |Y_{k,k}| / \sum_{k=1}^{N_k} \sum_{l=1}^{N_k} |Y_{k,l}|, \quad (1)$$

где $Y_{k,k}$ – подмножество внутренних связей k -й части графа;

$Y_{k,l}$ – подмножество внешних связей между частями G_k и G_l ;

N_k – общее количество частей, на которые разбит G .

Данный коэффициент показывает соотношение внутренних и внешних связей между участниками проекта. Очевидно, чем больше значение K_G при существующем разделении ролей между участниками, тем более эффективно распределены задачи между ними.

Предложенная модель распределения задач между участниками проекта *дополняет* эволюционно-инкрементную модель ЖЦ ПС. Она *способствует* структурированию активностей участников проекта, а также упорядочиванию связей между ними. Рассчитав коэффициент разбиения графа для нескольких вариантов распределения задач и выбрав из их числа наилучшее по критерию внешних связей распределение, можно *повысить продуктивность* команды разработчиков.

Во второй главе также предложен новый *метод организации процесса разработки ПС в условиях часто изменяющихся требований*. Суть его состоит в следующем:

- разработка ПС ведется на основе гибких методов;
- для определения работ ЖЦ и связей между ними используется предложенная эволюционно-инкрементная модель ЖЦ ПС;
- для определения ролей команды разработчиков используется модель MSFT (Microsoft Solutions Framework Team Model – модель команды каркаса решений Майкрософт);
- распределение задач между участниками проекта выполняется на основе предложенной модели распределения задач ЖЦ ПС;
- принцип создания архитектуры ПС, определяемый выбранным гибким методом, заменяется принципом прямого создания архитектуры на основе требований;
- осуществляется управление реализацией изменяющихся требований;
- реализуется принцип ориентации на быструю модификацию ПС на этапе сопровождения.

Принцип прямого создания архитектуры на основе требований состоит в том, что программная архитектура определяется как отображение требований верхнего уровня в компоненты программной реализации $F: R \rightarrow C$, которые связываются согласно зависимостям требований:

$$F = \left(\begin{array}{cccc} r_1 & r_2 & \dots & r_{n_r} \\ F(r_1) & F(r_2) & \dots & F(r_{n_r}) \end{array} \right), \quad (2)$$

где $F(r_i) \in C, i = \overline{1, n_r}$.

Управление реализацией изменяющихся требований сводится к поддержанию целостности (согласованности) кода компонента и того требования, которое он реализует. Любое изменение требования автоматически обнаруживает те компоненты, которые должны быть пересмотрены вследствие произведенной модификации. Для сигнализации о несогласованности кода требованиям предложено использовать ошибки или предупреждения на этапе сборки проекта, выдаваемые средой разработки. Задача определения порядка сборки компонентов сводится к получению ориентированного полугамильтонова графа $G(C)$. В главе 2 предложен алгоритм получения графа $G(C)$ на основе зависимостей компонентов. На рисунке 2 приведен возможный вариант зависимостей компонентов для некоторого ПС и соответствующий ему полугамильтонов граф (в этом примере компонент C_1 зависит от компонентов C_2 и C_3 , компонент C_2 зависит от C_4 и т.д., а сборка компонентов ПС происходит в порядке C_6, C_7, C_5 и т.д.).

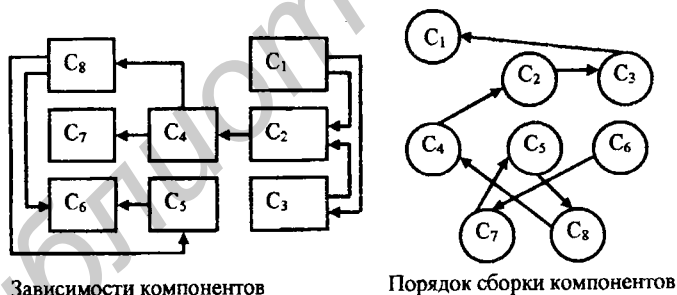


Рисунок 2 - Пример полугамильтонова графа для некоторого ПС

С целью быстрой модификации ПС на этапе сопровождения применяется шаблон проектирования MVC (Model-View-Controller – модель-представление-контроллер). Для формального определения этого шаблона задается объединение S элементов MVC (элемент MVC – это либо модель, определяющая данные, либо представление, определяющее способ

отображения данных, либо контроллер, определяющий поведение компонента) и задается отображение $L: C \rightarrow S$ множества компонентов C в множество S :

$$S = M \cup V \cup K = \{s | s \in M \text{ или } s \in V \text{ или } s \in K\}, \quad (3)$$

где M – множество моделей;
 V – множество представлений;
 K – множество контроллеров;

$$L: C \rightarrow S, \quad (4)$$

где C – множество компонентов ПС, причем $\forall c \in C \exists$ как минимум одно $s \in S$ и $\forall s \in S \exists$ одно и только одно $c \in C$.

За счет MVC, реализованного в отдельном компоненте, достигается легкость модификации данного компонента при изменении требований к нему или изменении тех компонентов, от которых он зависит.

Аналогами предложенного метода организации процесса разработки ПС являются метод, сочетающий экстремальное программирование и Scrum, а также метод, сочетающий гибкое моделирование и экстремальное программирование. Преимуществами предложенного метода перед ними являются устойчивость процесса разработки к изменению требований в произвольный момент времени и возможность быстрой модификации ПС после сдачи его в эксплуатацию.

В третьей главе предложена *иерархическая модель качества ПС*, позволяющая более полно учитывать особенности разработки на основе гибких методов при оценке качества ПС. Данная модель основана на характеристиках и подхарактеристиках стандарта СТБ ИСО/МЭК 9126:2003 (при этом показана связь этого стандарта с разрабатываемой в настоящее время серией международных стандартов ISO/IEC 25000), использует и адаптирует ряд метрик стандарта ISO/IEC TR 9126-2:2003 и предлагает набор собственных метрик.

Суть адаптации существующих метрик состоит в их приведении к относительным единицам и нормировании, чтобы значения метрик изменялись на отрезке $[0;1]$, причем чем ближе к 1, тем выше качество ПС. За счет адаптации все используемые в работе метрики соответствуют их свойствам и критериям обоснованности.

Предложенная модель позволяет оценивать внешнее качество ПС. Она представлена в виде множества Q :

$$Q = \{F, R, E, M\}, \quad (5)$$

где F – множество подхарактеристик функциональности;
 R – множество подхарактеристик надежности;
 E – множество подхарактеристик эффективности;
 M – множество подхарактеристик сопровождаемости.

Для оценки подхарактеристик *функциональности* ПС предложены следующие внешние метрики: приобретенная функциональность (позволяет установить влияние использования сторонних компонентов), точность вычислений (показывает, насколько точны результаты вычислений ПС), зависимость от внешних подсистем (показывает, насколько зависимо данное ПС от внешних подсистем). Разработанная модель внешнего качества для оценки функциональности ПС приведена на рисунке 3 (жирным шрифтом выделены предложенные метрики).



Рисунок 3 - Модель внешнего качества для оценки функциональности ПС

Разработанная модель внешнего качества для оценки надежности ПС приведена на рисунке 4 (жирным шрифтом выделены предложенные метрики, курсивом – адаптированные).

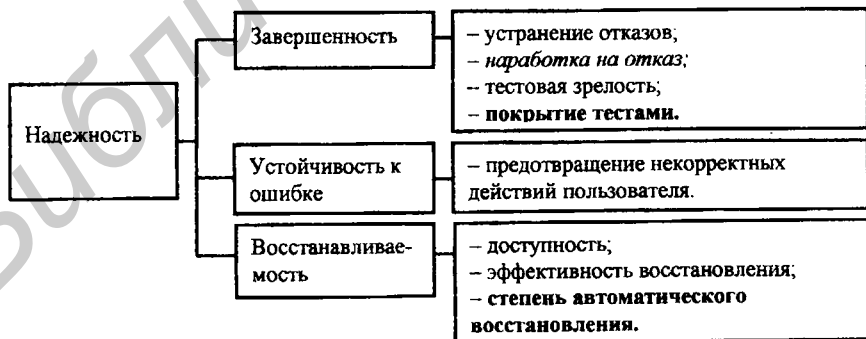


Рисунок 4 - Модель внешнего качества для оценки надежности ПС

Для оценки подхарактеристик *надежности* ПС предложены следующие внешние метрики: покрытие тестами (показывает, какова доля функциональных требований, для которых разработаны тесты) и степень автоматического восстановления (позволяет определить, насколько эффективно ПС может восстанавливать свою работоспособность без участия пользователя).

Для оценки подхарактеристик *эффективности* ПС предложены следующие внешние метрики: распределение длительных операций (позволяет определить равномерность распределения операций, требующих большего времени для выполнения, по компонентам ПС), соотношение времени отклика и затрат памяти (позволяет определить, насколько оптимально соотношены время отклика и потребляемая память). Модель внешнего качества для оценки эффективности ПС приведена на рисунке 5 (жирным шрифтом выделены предложенные метрики, курсивом – адаптированные).



Рисунок 5 - Модель внешнего качества для оценки эффективности ПС

Для оценки подхарактеристик *сопровожаемости* ПС предложены следующие внешние метрики: повторное использование решений (показывает интенсивность использования уже опробованных в других проектах решений), сложность конфигурирования (показывает, насколько сложен процесс конфигурирования ПС), способность к открытию интерфейса прикладного программирования (показывает, насколько легко ПС может предоставлять собственный интерфейс прикладного программирования на уровне компонентов), время обновления у пользователя (показывает, насколько быстро проходит обновление ПС), успешность изменений (показывает, сколько новых отказов появилось в результате модификации ПС по отношению к общему числу отказов, включая устраненные). Разработанная модель внешнего качества для оценки сопровождаемости ПС приведена на рисунке 6 (жирным шрифтом выделены предложенные метрики, курсивом – адаптированные).

С целью оценки качества ПС на основе приведенных метрик функциональности, надежности, эффективности и сопровождаемости в третьей главе предложен модифицированный метод оценки качества ПС.



Рисунок 6 - Модель внешнего качества для оценки сопровождаемости ПС

В четвертой главе приведено описание разработанных *инструментальных средств* для эффективного применения на практике предложенного метода организации процесса разработки ПС в условиях часто изменяющихся требований. Разработано инструментальное средство, позволяющее получать набор связанных проектов Microsoft Visual Studio автоматически на основе требований, представленных в СУТ IBM Rational DOORS. Разработана надстройка к Microsoft Visual Studio, реализующая механизм управляемой сборки ПС. Разработана программная платформа, обеспечивающая простоту и высокую скорость модификации ПС на этапе сопровождения.

В четвертой главе также приведены результаты *практических исследований*, доказывающие преимущества предложенных моделей, метода организации процесса разработки и инструментальных средств перед аналогами. Данные модели, метод и инструментальные средства были использованы при разработке:

- ПС управления финансовыми портфелями;
- модуля бухгалтерского учета ПС электронной торговли синдицированными кредитами;
- ПС мониторинга и анализа движения транспортных средств;
- ПС биржевой торговли на основе динамически определяемых стратегий.

Указанные ПС сравнивались с ПС, имеющими схожую функциональность и разработанными примерно за такое же время. В результате экспериментов установлено, что качество ПС при использовании предложенных моделей, метода и инструментальных средств более чем на 15 % выше по сравнению с

аналогами. Усредненная оценка качества по отдельным характеристикам для экспериментальных ПС приведена на рисунке 7.

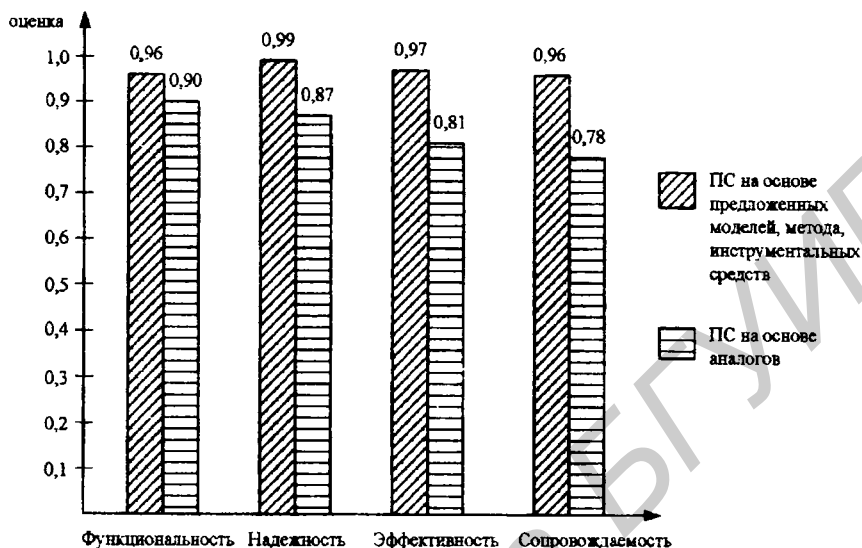


Рисунок 7 – Сравнение характеристик качества экспериментальных ПС

На основе проведенных экспериментов определены условия и даны рекомендации по применению на практике предложенных моделей и метода организации процесса разработки. Эти условия и рекомендации разделены на организационные и технические. Главным организационным ограничением является требование к масштабу проекта: реализуемый проект должен быть небольшим по масштабу, т.е. команда разработчиков численностью до 20 человек должна быть способна реализовать его за 3–4 месяца (классификация проектов приведена в стандарте ГОСТ Р ИСО/МЭК ТО 12182-2002). Техническими условиями и рекомендациями являются следующие: требования должны быть структурированы по функциям ПС (пример соответствующей спецификации требований приведен в стандарте IEEE 830:1998); для управления требованиями рекомендуется использовать ПС IBM Rational DOORS или ПС аналогичного функционального назначения; предложенные инструментальные средства позволяют разрабатывать настольные ПС для платформы Microsoft .NET, при этом средой разработки является Microsoft Visual Studio.

ЗАКЛЮЧЕНИЕ

Основные научные и практические результаты исследования

Данная диссертационная работа предлагает решение актуальной проблемы *повышения качества ПС и оценки качества ПС с учетом специфики изменяющихся требований*. Предложенные модели и метод предоставляют

комплексное решение для разработки в условиях изменяющихся требований, начиная с определения процессов ЖЦ и заканчивая оценкой качества ПС.

В результате выполнения диссертационной работы решены следующие задачи:

1. Проведен анализ моделей ЖЦ ПС, методов разработки ПС, моделей качества ПС при их применении в условиях часто изменяющихся требований. Установлено, что существующие модели ЖЦ ПС не позволяют создавать качественные ПС в таких условиях из-за определения требований в начале ЖЦ (каскадные и инкрементные модели) или длины итерации и сложной структуры модели (эволюционные модели) [8–А]. Проанализированы гибкие методы разработки (FDD, XP, Scrum, DSDM, ASD и др.), выявлены их недостатки: отсутствие возможности модифицировать требования к ПС внутри инкремента разработки, неуправляемость внесения изменений в ПС, отсутствие механизмов быстрой модификации ПС на этапе сопровождения [6–А, 16–А]. Проанализированы базовые модели качества ПС (модели МакКола, Бозма, FURPS/FURPS+, Дроми), а также модели качества по стандартам ISO/IEC 9216-1:2001 (СТБ ИСО/МЭК 9126:2003) и ISO/IEC 25011. Выявлены их недостатки: использование характеристик и метрик качества, которые не отвечают современным тенденциям разработки ПС; выстраивание неверной иерархии характеристик и подхарактеристик качества; отсутствие метрик, позволяющих в полной мере учесть особенности разработки на основе гибких методов [11–А, 12–А].

2. Разработана эволюционно-инкрементная модель ЖЦ ПС для гибких методов. В качестве основных выделены процессы заказа, разработки, эксплуатации и сопровождения. Определены связи между ними. В итеративном процессе разработки предложено использовать два вида итераций: внешние (реализуют эволюционную стратегию разработки) и внутренние (реализуют инкрементную стратегию разработки) [8–А, 17–А]. Разработана модель распределения задач ЖЦ между участниками проекта. Распределение задач представлено в виде ориентированного графа, вершинами которого являются члены команды разработчиков, а дугами – их связи. Для сравнения различных вариантов распределения задач и выбора достаточно эффективной в заданных условиях организации участников проекта предложено использовать коэффициент разбиения графа, который показывает соотношение внутренних и внешних связей между ними [9–А].

3. Предложен метод организации процесса разработки ПС в условиях часто изменяющихся требований. Метод базируется на применении эволюционно-инкрементной модели ЖЦ ПС и соответствующей ей модели распределения задач между участниками проекта, а также предлагает набор принципов разработки ПС и их конкретную реализацию. Разработан механизм управления реализацией изменяющихся требований на основе управляемой сборки ПС, предложен алгоритм определения порядка сборки компонентов ПС [2–А, 3–А, 4–А, 6–А, 7–А, 10–А, 13–А, 18–А, 19–А, 20–А, 21–А, 22–А]. С целью программной поддержки предложенного метода разработаны: инструментальное средство автоматического создания архитектуры на основе

требований, надстройка к Microsoft Visual Studio для обеспечения согласованности требований и их реализации, программная платформа для обеспечения простоты и высокой скорости модификации ПС на этапе сопровождения [5-А, 15-А].

4. Предложена иерархическая модель качества ПС, позволяющая более полно учитывать особенности разработки на основе гибких методов при оценке качества ПС. В модель входят такие характеристики, как функциональность, надежность, эффективность и сопровождаемость. Предложены новые внешние метрики качества: для оценки подхарактеристик функциональности – приобретенная функциональность, точность вычислений, зависимость от внешних подсистем; надежности – покрытие тестами, степень автоматического восстановления; эффективности – соотношение времени отклика и затрат памяти, распределение длительных операций; сопровождаемости – повторное использование решений, сложность конфигурирования, способность к открытию интерфейса прикладного программирования, время обновления у пользователя, успешность изменений [1-А, 11-А, 12-А, 14-А, 23-А].

5. С целью проверки на практике предложенных моделей, метода и инструментальных средств проведен ряд экспериментов по разработке как промышленных ПС (ПС управления финансовыми портфелями, модуль бухгалтерского учета ПС электронной торговли синдицированными кредитами, ПС биржевой торговли на основе динамически определяемых стратегий), так и экспериментальных ПС (ПС мониторинга и анализа движения транспортных средств). В результате данных экспериментов установлено, что качество ПС при использовании предложенных моделей, метода и инструментальных средств более чем на 15 % выше по сравнению с аналогами [6-А].

Предложенные в диссертационной работе модели и методы допускают дальнейшее развитие и улучшение с целью повышения их практической пользы.

Рекомендации по практическому использованию результатов

Результаты диссертационной работы позволяют улучшить качество ПС, разрабатываемых на основе гибких методов в условиях часто изменяющихся требований.

Исследования, проведенные в диссертационной работе, опираются на передовой мировой опыт в области разработки ПС, учитывают тенденции развития информационных технологий. Результаты исследований могут быть применены при разработке ПС на основе гибких методов, когда велик риск изменения требований к ПС, а сроки реализации ограничены.

Результаты диссертационной работы внедрены в производственный процесс ООО «Численные методы», процесс сопровождения ПС филиала ОАО МАЗ «Минский рессорный завод», процесс сопровождения ПС ЗАО «Дарасофт» и учебный процесс БГУИР.

СПИСОК ПУБЛИКАЦИЙ СОИСКАТЕЛЯ ПО ТЕМЕ ДИССЕРТАЦИИ

Статьи в научных журналах

1–А. Бахтизин, В.В. Разработка требований к эффективности программных средств / В.В. Бахтизин, Л.А. Глухова, С.Н. Неборский // Известия Белорусской инженерной академии. – 2005. – № 1 (19). – С. 31–35.

2–А. Бахтизин, В.В. Концепция управления требованиями к программным средствам / В.В. Бахтизин, С.Н. Неборский // Доклады Белорусского государственного университета информатики и радиоэлектроники. – 2006. – № 3 (15). – С. 144–148.

3–А. Бахтизин, В.В. Создание управляемой программной архитектуры / В.В. Бахтизин, С.Н. Неборский // Программные продукты и системы. – 2006. – № 3 (75). – С. 2 – 5.

4–А. Бахтизин, В.В. Технология создания программной архитектуры на платформе .NET / В.В. Бахтизин, С.Н. Неборский // Информатизация образования. – 2006. – № 1 (42). – С. 45–52.

5–А. Бахтизин, В.В. Использование метаданных .NET для автоматизации процесса управления требованиями / В.В. Бахтизин, С.Н. Неборский, О.Н. Бачило // Инженерный вестник : информатика, радиофизика, управление. – 2006. – № 1 (21). – С. 160–162.

6–А. Бахтизин, В.В. Обеспечение качества программных средств в условиях изменяющихся требований / В.В. Бахтизин, С.Н. Неборский // Проблемы программирования. – 2008. – № 2–3. – С. 173–180.

7–А. Бахтизин, В.В. Разработка программных средств на основе гибких методов / В.В. Бахтизин, С.Н. Неборский // Программные продукты и системы. – 2008. – № 2 (82). – С. 44–47.

8–А. Бахтизин, В.В. Модель жизненного цикла программных средств при разработке на основе гибких методов / В.В. Бахтизин, С.Н. Неборский // Доклады Белорусского государственного университета информатики и радиоэлектроники. – 2008. – № 8 (38). – С. 75–81.

9–А. Бахтизин, В.В. Модель распределения задач при разработке программных средств на основе гибких методов / В.В. Бахтизин, С.Н. Неборский // Информатизация образования. – 2009. – № 2. – С. 74–84.

10–А. Неборский, С.Н. Модель ролей команды разработчиков программных средств на основе гибких методов / С.Н. Неборский // Доклады Белорусского государственного университета информатики и радиоэлектроники. – 2009. – № 2 (40). – С. 73–79.

Статьи в сборниках материалов научных конференций

11–А. Бахтизин, В.В. Управление качеством разработки обучающихся программных продуктов по критерию эффективности / В.В. Бахтизин, С.Н. Неборский // Дистанционное обучение – образовательная среда XXI века: материалы IV Международной научно-методической конференции, Минск, 10 –

12 ноября 2004 г. / БГУИР; под ред. М.П. Батуры [и др.]. – Минск, 2004. – С. 284–287.

12–А. Бахтизин, В.В. Дидактическая оптимизация обучающих программных средств / В.В. Бахтизин, С.Н. Неборский // Инженерно-педагогическое образование : проблемы и пути развития : международная научно-методическая конференция, Минск, 7 – 8 октября 2004 г. : сборник научных статей : в 2 ч. / МГВРК; под ред. Н.А. Цырельчука [и др.]. – Минск, 2004. – Ч. 1. – С. 86–89.

13–А. Бахтизин, В.В. Реализация концепции управления требованиями к программным средствам / В.В. Бахтизин, С.Н. Неборский // Дистанционное обучение – образовательная среда XXI века: материалы V Международной научно-методической конференции, Минск, 10 – 11 ноября 2005 г. / БГУИР; под ред. М.П. Батуры [и др.]. – Минск, 2005. – С. 249–252.

14–А. Бахтизин, В.В. Повышение эффективности разработки программных средств / В.В. Бахтизин, С.Н. Неборский // Современная радиоэлектроника : научные исследования, подготовка кадров : международная научно-практическая конференция, Минск, 20 – 21 апреля 2006 г. : сборник материалов : в 3 ч. / МГВРК; под ред. Н.А. Цырельчука [и др.]. – Минск, 2006. – Ч. 1. – С. 190–192.

15–А. Неборский, С.Н. Обеспечение легкой модификации .NET-приложений / С.Н. Неборский // Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях : материалы X Республиканской научной конференции студентов и аспирантов, Гомель, 12 – 14 марта 2007 г. / ГГУ; под ред. Д.Г. Лина [и др.]. – Гомель, 2007. – С. 340–341.

16–А. Бахтизин, В.В. Выбор метода разработки программных средств в контексте изменяющихся требований / В.В. Бахтизин, С.Н. Неборский // Современная радиоэлектроника : научные исследования, подготовка кадров: международная научно-практическая конференция, Минск, 20 – 21 апреля 2006 г. : сборник материалов : в 3 ч. / МГВРК; под ред. Н.А. Цырельчука [и др.]. – Минск, 2007. – Ч. 2. – С. 21–23.

17–А. Бахтизин, В.В. Итеративный подход к разработке программных средств / В.В. Бахтизин, С.Н. Неборский // Дистанционное обучение – образовательная среда XXI века : материалы VI Международной научно-методической конференции, Минск, 22 – 23 ноября 2007 г. / БГУИР; под ред. М.П. Батуры [и др.]. – Минск, 2007. – С. 214–215.

18–А. Бахтизин, В.В. Организационная структура команды разработчиков программных средств / В.В. Бахтизин, С.Н. Неборский // Современные информационные компьютерные технологии : сборник научных статей / ГрГУ; под ред. А.М. Кадана [и др.]. – Гродно, 2008. – С. 18–21.

19–А. Бахтизин, В.В. Роли команды разработчиков программных средств / В.В. Бахтизин, С.Н. Неборский // Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях : материалы XI Республиканской научной конференции

студентов и аспирантов, Гомель, 17 – 19 марта 2008 г. / ГГУ; под ред. О.М. Демиденко [и др.]. – Гомель, 2008. – С. 85–86.

20–А. Неборский, С.Н. Архитектура компонентов программного средства / С.Н. Неборский // Современная радиоэлектроника : научные исследования и подготовка кадров : международная научно-практическая конференция, Минск, 23 – 24 апреля 2008 г. : сборник материалов : в 3 ч. / МГВРК; под ред. Н.А. Цырельчука [и др.]. – Минск, 2008. – Ч. 2. – С. 54–55.

21–А. Неборский, С.Н. Разработка обучающих программных средств на основе гибких методов / С.Н. Неборский // Материалы IV Международной научно-практической конференции «Образование и наука без границ», Прага, 7 – 15 декабря 2008 г. : Т. 17 : математика, физика, новые информационные технологии / под ред. С. Горняка [и др.]. – Прага, 2008. – С. 56–58.

22–А. Бахтизин, В.В. Отношения «принципал-агент» в команде разработчиков программных средств / В.В. Бахтизин, С.Н. Неборский // Информационные системы и технологии = Informational systems and technologies : (IST'2009) : материалы V международной конференции-форума (Минск, 16 – 17 ноября 2009 г.) : в 2 ч. / [редколлегия: Н. И. Листопад и др.]. – Минск, 2009. – Ч. 1. – С. 267–270.

23–А. Неборский, С.Н. Оценка качества программных средств при разработке на основе гибких методов / С.Н. Неборский // Информационные системы и технологии = Informational systems and technologies : (IST'2010) : материалы VI международной конференции-форума (Минск, 24 – 24 ноября 2010 г.) / БГУ; под ред. А.Н. Вараскина [и др.]. – Минск, 2010. – С. 364–366.

С'Нед

Няборскі Сяргей Мікалаевіч

МАДЭЛІ, МЕТАДЫ І СРОДКІ РАСПРАЦОЎКІ І АЦЭНКІ ЯКАСЦІ ПРАГРАМНАГА ЗАБЕСПЯЧЭННЯ ВА ЎМОВАХ ЗМЕНЛІВЫХ ПАТРАБАВАННЯЎ

Ключавыя словы: забеспячэнне якасці, ацэнка якасці, гнуткія метады, мадэль якасці, арганізацыя працэсу распрацоўкі.

Мэтай дысертацыйнай працы з'яўляецца стварэнне мадэляў і метадаў распрацоўкі і ацэнкі якасці праграмных сродкаў, якія ўлічваюць спецыфіку часта зменлівых патрабаванняў. Аб'ектам даследавання з'яўляюцца працэс распрацоўкі і ацэнка якасці праграмных сродкаў ва ўмовах зменлівых патрабаванняў. Прадметам даследавання з'яўляюцца гнуткія метады распрацоўкі праграмных сродкаў.

Дысертацыйная праца змяшчае комплекснае рашэнне для распрацоўкі праграмных сродкаў ва ўмовах часта зменлівых патрабаванняў. Прапанавана эвалюцыйна-інкрыментная мадэль жыццёвага цыкла праграмных сродкаў для гнуткіх метадаў, якая вызначае працэсы, работы і задачы жыццёвага цыкла, іх паслядоўнасць і ўзаемасувязь. У дадатак да яе прапанавана мадэль размеркавання задач паміж удзельнікамі праекта. Прапанаваны метады арганізацыі працэсу распрацоўкі праграмных сродкаў, які дазваляе павысіць якасць праграмных сродкаў ва ўмовах часта зменлівых патрабаванняў. Распрацаваны інструментальныя сродкі для падтрымкі ладзенага метаду. Прапанавана мадэль якасці праграмных сродкаў, якая дазваляе больш поўна ўлічваць асаблівасці распрацоўкі на аснове гнуткіх метадаў пры ацэнцы якасці праграмных сродкаў, а таксама мадыфікацыя метаду ацэнкі якасці.

Выкарыстанне прапанаваных мадэляў, метадаў і інструментальных сродкаў дазваляе ствараць больш якасныя праграмныя сродкі пры распрацоўцы на аснове гнуткіх метадаў ва ўмовах часта зменлівых патрабаванняў і ўлічваць спецыфіку распрацоўкі ў такіх умовах пры ацэнцы якасці праграмных сродкаў.

РЕЗЮМЕ

Неборский Сергей Николаевич

МОДЕЛИ, МЕТОДЫ И СРЕДСТВА РАЗРАБОТКИ И ОЦЕНКИ КАЧЕСТВА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В УСЛОВИЯХ ИЗМЕНЯЮЩИХСЯ ТРЕБОВАНИЙ

Ключевые слова: обеспечение качества, оценка качества, гибкие методы, модель качества, организация процесса разработки.

Целью диссертационной работы является создание моделей и методов разработки и оценки качества программных средств, учитывающих специфику часто изменяющихся требований. Объектом исследования являются процесс разработки и оценка качества программных средств в условиях изменяющихся требований. Предметом исследования являются гибкие методы разработки программных средств.

Данная работа содержит комплексное решение для разработки программных средств в условиях часто изменяющихся требований. Предложена эволюционно-инкрементная модель жизненного цикла программных средств для гибких методов, которая определяет процессы, работы и задачи жизненного цикла, их последовательность и взаимосвязь. В дополнение к ней предложена модель распределения задач между участниками проекта. Предложен метод организации процесса разработки программных средств, который позволяет повысить качество программных средств в условиях часто изменяющихся требований. Разработаны инструментальные средства для поддержки данного метода. Предложена модель качества программного средства, позволяющая более полно учитывать особенности разработки на основе гибких методов при оценке качества программных средств, а также модификация метода оценки качества.

Применение предложенных моделей, методов и инструментальных средств позволяет создавать более качественные программные средства при разработке на основе гибких методов в условиях часто изменяющихся требований и учитывать специфику разработки в таких условиях при оценке качества программных средств.

SUMMARY

Niaborski Siarhei Mikalaeovich

MODELS, METHODS AND TOOLS FOR SOFTWARE DEVELOPMENT AND ASSESSMENT IN CHANGING REQUIREMENTS CONDITIONS

Keywords: quality assurance, quality assessment, agile methods, quality model, organizing development process.

This research aims to develop models, methods and products which allow to assure and estimate software quality considering changing requirements conditions. The research object is definition of software development process in changing requirements environment. The research subject is agile development methods.

The goal of the research is to provide a comprehensive solution which helps to develop software with agile methods and to estimate its quality better. The first thing offered is a software lifecycle model which combines evolutionary and incremental strategies for agile methods. The model defines lifecycle processes, their sequence and interconnections. A scope of tasks which are solved within each process is also defined there. In addition to software lifecycle model, a model to distribute tasks between team members is introduced. Having a software lifecycle model and task distributing model, a new method of organizing development process is offered. The method allows to assure software quality when software requirements are changed frequently, and it is needed to quickly update the software at any stage of a lifecycle. To help developers use this method, special tools are offered: a software architecture generator, add-in to manage changing requirements, and software platform. Eventually, a software quality model is introduced and a modified method for its estimation is offered.

When used in a production environment, the proposed models, methods and tools increase software quality and allow to consider specific features of changing requirements conditions.

НЕБОРСКИЙ Сергей Николаевич

**МОДЕЛИ, МЕТОДЫ И СРЕДСТВА РАЗРАБОТКИ И ОЦЕНКИ
КАЧЕСТВА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В УСЛОВИЯХ
ИЗМЕНЯЮЩИХСЯ ТРЕБОВАНИЙ**

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

АВТОРЕФЕРАТ
диссертации на соискание ученой степени
кандидата технических наук

Подписано в печать 25.04.2011.	Формат 60x84 ¹ / ₁₆ .	Бумага офсетная.
Гарнитура «Таймс».	Отпечатано на ризографе.	Усл. печ. л. 1,63.
Уч.-изд. л. 1,5.	Тираж 60 экз.	Заказ 258.

Издатель и полиграфическое исполнение: учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники»
ЛИ №02330/0494371 от 16.03.2009. ЛП №02330/0494175 от 03.04.2009
220013, Минск, П. Бровки, 6