

УДК 65.012.123

А.В. Баби́ч

Полтавський політехнічний коледж НТУ «ХПІ», Полтава

УПРАВЛЕНИЕ ПРОЦЕССОМ РАЗРАБОТКИ ПРОГРАММНОЙ СИСТЕМЫ С ИСПОЛЬЗОВАНИЕМ МЕТОДОЛОГИИ P-MODELING FRAMEWORK И ОБРАТНОЙ СЕМАНТИЧЕСКОЙ ТРАССИРОВКИ

Ставится и решается задача оптимизации управления разработкой программной системы (ПС) с целью обеспечения качества создаваемого программного продукта. Сформирован метод управления разработкой ПС, основанный на совершенствовании технологий P-MF и обратной семантической трассировки. Процесс разработки представлен в виде итеративного процесса с контролем выполнения требований к новому проекту по созданию ПС. Для оценки близости описаний проекта используются качественные оценки экспертов, которые сравниваются с контрольным представлением близости, задаваемым командой проектировщиков. Предложенный подход позволяет обеспечить качество разрабатываемого программного продукта, снижает риски проектирования и сроки разработки ПС.

Ключевые слова: *качество программного продукта, методология P-MF, RST, обратная семантическая трассировка, лексикографическое упорядочивание вариантов, экспертное оценивание.*

Введение

Повышение сложности и наукоемкости разрабатываемых программных продуктов, ужесточение требований к качеству программных систем (ПС) привело к необходимости совершенствования процесса управления разработкой программных систем.

Современная методология P-Modeling Framework (P-MF) направлена на улучшение процесса проектирования ПС, но требует дальнейшего совершенствования для обеспечения качества конечного программного продукта [1].

Поэтому актуальна тема предлагаемой публикации, в которой предлагается итеративный процесс управления разработкой для обеспечения качества программных систем с использованием P-MF и ее составляющей – обратной семантической трассировки [2].

Постановка задачи исследования

Методология P-MF представляет собой набор принципов, методов и инструментов, которые позволяют оптимизировать жизненный цикл (ЖЦ) программных систем, повышают эффективность процесса и сокращают сроки разработки системы.

Обратная семантическая трассировка (RST) обеспечивает контроль проектирования системы и направлена на обеспечение качества создаваемой программной системы.

Формирование формального метода, который представляет P-MF и ее составляющую – RST, в форме процесса с постоянным контролем качества проектирования подробно рассматривается в данной работе.

Решение задачи исследования

Для обеспечения качества проектирования программной системы предлагаем метод, основанный на методологии P-MF.

Метод состоит из следующих этапов:

1. На основании исходного описания (ИО) программной системы с помощью разработчиков базовой платформы (БП) новой программной системы (НПС), создается модель в виде множества диаграмм (классов, прецедентов, последовательностей и т.д.) компонентной архитектуры системы.

2. На втором этапе, осуществляется обратная семантическая трассировка [2], в которой разработчики, разрабатывающие программную систему в рамках нового проекта (НП), подробно не ознакомленные с требованиями и описанием НПС, должны на основе моделей первого этапа восстановить первоначальное описание (ВО).

3. На следующем этапе проводится проверка соответствия исходного описания и восстановленного первоначального описания, полученных разработчиками БП и НП. Если степень совпадения описаний НПС (ИО и ВО) разработчиков БП и НП достаточно велика (вводится критерий близости), то в соответствии с процедурой RST принимается решение о переходе к разработке НПС.

В противном случае (большое несоответствие и отсутствие близости описаний ИО и ВО), осуществляется доработка модели, полученной от разработчиков БП. Если, после доработки модели и повторной сессии RST, соответствие ИО и ВО не наблюдается, то осуществляется уточнение требований к ПС и, соответственно, исходного описания, после чего процесс обратной семантической трас-

сировки повторяется еще раз (очередная итерация процесса).

В предлагаемом методе, основанном на P-MF и ее составляющей – RST, большое внимание уделяется оценке близости (совпадения) описаний НПС разработчиков БП и НП, на основании которой принимается решение о начале непосредственных действий по разработке новой программной системы.

Для выбора метрики оценки близости формируемых описаний новой программной системы введем ряд представлений (признаков), которые связаны с:

- диаграммами классов – P_1 ;
- диаграммами прецедентов – P_2 ;
- диаграммами последовательностей – P_3 ;
- диаграммами компонентов – P_4 ;
- диаграммами развертывания – P_5 ;
- текстами с описанием НПС – P_6 .

Для оценки близости описаний НПС для разработчиков БП и НП по каждому P_i признаку можно воспользоваться как количественными, так и качественными оценками. Основные метрики, которые используются при определении меры близости объектов используют количественные представления признаков [3].

В качестве метрики определения меры близости описаний ИО и ВО, при проектировании новой программной системы, учитывая, преимущественно, сложное представление признаков, будем в работе использовать качественную оценку близости в виде значений лингвистической переменной I_{P_i} :

$$I_{P_i} = \begin{cases} A - \text{высокая степень соответствия} \\ \text{описаний для } i\text{-го признака,} \\ B - \text{хорошая степень соответствия,} \\ C - \text{удовлетворительная степень} \\ \text{соответствия,} \\ D - \text{соответствие не наблюдается.} \end{cases}$$

Пусть, в самом начале процедуры оценивания близости моделей, сформированы возможные (допустимые) расхождения ИО и ВО новой программной системы разработчиков БП и НП по каждому признаку, которые, в свою очередь, с учетом важности расположения в ряд, в начале которого находится самый важный признак.

Например,

$$P_1 \succ P_2 \succ P_3 \succ P_4 \succ P_5 \succ P_6.$$

Тогда, учитывая значения I_{P_i} , полученное от экспертов по каждому i -му признаку, можно сформировать контрольное «слово», состоящее из букв латинского алфавита, которое является основой для оценивания близости описаний новой про-

граммной системы, созданных разработчиками БП и НП.

Например, контрольное «слово», с учетом множества упорядоченных признаков, имеет вид:

$$B, A, B, A, B, C.$$

Пусть в результате проверки соответствия описаний новой программной системы и оценивания экспертами всех признаков (P_1, \dots, P_6) получим «слово»:

$$1. A, A, B, A, B, B.$$

Лексикографически упорядочим список, состоящий из двух «слов» (одно из них контрольное):

$$1. A, A, B, A, B, B$$

$$K. B, A, B, A, B, C.$$

В этом списке первое слово находится выше контрольного.

В данном случае, близость описаний новой программной системы наблюдается с учетом всех признаков. Поэтому можно приступить к этапу разработки НПС.

Пусть, в результате экспертного оценивания, получено «слово», для оценки близости описаний новой программной системы, в виде:

$$2. B, A, C, B, B, B.$$

Тогда, после лексикографического упорядочивания «слов», получим:

$$K. B, A, B, A, B, C,$$

$$2. B, A, C, B, B, B.$$

В этом случае, из-за того, что второе «слово» находится ниже контрольного можно считать, что близости не наблюдается, поэтому необходимо вернуться на шаг назад, и перейти к доработке модели ПС.

Если для оценки близости описаний используется группа экспертов, то для ее формирования необходимо, в начале, оценивать компетентность экспертов. При этом используются различные методы оценивания самих экспертов, например, бальные оценки профессиональной компетентности экспертов [3].

Определение количества и персонального состава экспертной группы, является, в настоящее время, слабо формализованной задачей, которая зависит от целей оценивания и особенностей предметной области (в данном случае, предметная область программной инженерии). Установление оптимального количества экспертов в группе является сложной задачей. Если экспертов много, то в группу, с высокой вероятностью, попадают некомпетентные эксперты. Если же экспертов мало, то результат экспертизы существенно зависит от кон-

кретных лиц, попавших в эксперты. Для нахождения оптимального количества экспертов можно использовать метод, основанный на поиске нижней и верхней границ числа экспертов (N_{\min} , N_{\max}) [3].

Таким образом, численность экспертов в группе устанавливается в пределах: $N_{\min} \leq N \leq N_{\max}$. В зависимости от величины измерения средней ошибки ε , минимальную численность экспертной группы можно, также, определять по такой формуле [3]:

$$N_{\min} = 0,5 \left(\frac{3}{\varepsilon} + 5 \right).$$

Численность экспертов в группе можно определить и на основе выборочного наблюдения [3]:

$$N = \frac{P(1-P)}{\Delta_p^2} K_c^2,$$

где N – число экспертов в группе;

P – веса экспертов, которые удовлетворяют признакам, установленным организаторами экспертизы (например, наличие не менее 3 лет практического опыта);

K_c – критерий Стьюдента при заданном уровне вероятности (интервал доверия);

Δ_p – средняя граничная ошибка.

После формирования группы экспертов осуществляется экспертное оценивание близости описаний НПС (ИО и ВО) разработчиков БП и НП каждым экспертом.

В результате формируется список «слов» отдельных экспертов, который необходимо сравнить с контрольным «словом».

Для сравнения удобно вначале лексикографически упорядочить множество «слов», полученных от экспертов.

Далее необходимо определить расположение контрольного «слова» относительно списка «слов», полученных с помощью экспертов.

Возможны следующие варианты расположения контрольного «слова» относительно множества «слов» экспертов:

1. Контрольное «слово» находится ниже списка «слов» экспертов. Например,

A, A, B, A, B, B

B, A, B, A, B, B

K. B, A, B, A, B, C

В этом случае принимается решение о близости ИО и ВО с учетом мнения всех экспертов и значит разрешения начала процесса разработки НПС.

2. Пусть «слова» экспертов находятся ниже контрольного «слова»:

K. B, A, B, A, B, C

B, A, C, B, B, B

B, B, A, C, B, B

Тогда принимается решение об отсутствии близости ИО и ВО. После чего, начинается процесс доработки модели.

3. Возможна ситуация, когда контрольное «слово» попадает в список множества «слов», которые предложили эксперты. В этом случае можно воспользоваться двумя подходами.

Первый подход связан с учетом важности (веса) экспертов.

Тогда оценка близости наиболее «важного» (важных) экспертов сравнивается с контрольным «словом» и принимается решение о близости ИО и ВО.

Второй подход связан с формированием групповой оценки экспертов по каждому признаку описания НПС.

Достаточно удобной и часто используемой процедурой формирования согласованной групповой оценки является метод Дельфи [3].

Процедура формирования групповой оценки по методу Дельфи заключается в анкетировании экспертов в несколько туров с обработкой результатов в каждом туре и информированием экспертов об этих результатах. На практике чаще всего ограничиваются четырьмя турами. В первом туре эксперты дают свои ответы без аргументирования. Ответы обрабатываются с целью выделения среднего и крайних мнений. Эти мнения экспертам сообщаются, и проводится второй тур опроса, в ходе которого они пересматривают и при желании изменяют ответы, данные в первом туре. При этом эксперты должны объяснить, почему они изменили или не изменили ответы.

Полученные после второго тура новые средние и крайние мнения, а также вся аргументация с сохранением анонимности сообщаются экспертам, и проводится третий тур опроса, в процессе которого эксперты снова пересматривают ответы и аргументируют свое решение. Последующие туры аналогичны.

Обычно, после третьего или четвертого тура опроса ответы экспертов перестают изменяться, что и является сигналом к прекращению опросов. Такая процедура позволяет экспертам учесть обстоятельства, которыми они пренебрегали или о которых не были осведомлены. По решению аналитика обоснование могут представлять лишь эксперты, ответы которых сильно отличаются от мнения большинства.

Организация экспертизы, связанная с разработкой НПС состоит из следующих основных этапов:

- предварительная ориентировка экспертов, включающая в себя формулирование проблемы, связанной с проектом по созданию новой программной системы и инструктаж экспертов;

- формулирование вопросов для экспертов в виде, требующем понимания экспертов содержания проекта по созданию новой программной системы.

Обмен информацией между экспертами проводится с соблюдением следующих правил:

- экспертам предоставляется подобранная руководителем проекта новой программной системы сводная информация относительно требований проекта;

- создаются условия, стимулирующие полное использование всеми экспертами предоставленной информации в виде описаний новой программной системы;

- разработка способов и порядка информационного обмена экспертов, включая обмен аргументацией;

- разработка процедур контроля промежуточных и окончательных оценок результатов.

Полученное «слово», на основании групповых оценок экспертов, сравнивается с контрольным «словом» и принимается решение относительно близости исходного описания и ВО разработчиков базовой платформы и нового проекта, а также решение о начале разработки новой программной системы.

Выводы

Предложенный метод целесообразно использовать не только на этапе формирования архитектуры программной системы, но и на последующих этапах процесса разработки новой программной системы, что позволяет контролировать качество создаваемых проектных артефактов, и в случае необходимости, вернуться к предыдущим этапам и повторить необходимые шаги с целью обеспечения соответствующего уровня качества конечного программного продукта.

Список литературы

1. *Using Reverse Semantic Traceability for Quality Control in Agile MSF-based Projects [Electronic resource] / K. Zhereb, V. Pavlov, A. Doroshenko, V. Sergienko // International Software & Productivity Engineering Institute (INTSPEI). – Moscow, Russia: 4th Software Engineering Conference, October 23, 2008: Proceedings. – Mode of access: WWW.URL: http://2008.cce-secr.org/ru/etc/secr2008_konstantin_zhereb_intspei_using_rst.pdf.*
2. *Pavlov V.L. Applying Pantomime and Reverse Engineering Techniques in Software Engineering Education / Pavlov V.L., Boyko N., Babich A., Kuchaiev O., Busygin S. // 37th ASEE/IEEE Frontiers in Education Conference. – Milwaukee, Wisconsin, USA, 2007. – P. TIE-1-TIE-5.*
3. *Интроспективный анализ. Методы и средства экспертного оценивания [Текст]: моногр. / В.В. Крючковский, Э.Г. Петров, Н.А. Соколова, В.Е. Ходаков; под. ред. Э.Г. Петрова. – Херсон: Гринь Д.С., 2011. – 168 с.*

Поступила в редколлегию 1.06.2016

Рецензент: д-р техн. наук, проф. И.В. Шостак, Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ», Харьков.

КЕРУВАННЯ ПРОЦЕСОМ РОЗРОБКИ ПРОГРАМНОЇ СИСТЕМИ З ЗАСТОСУВАННЯМ МЕТОДОЛОГІЇ P-MODELING FRAMEWORK ТА ЗВОРОТНОГО СЕМАНТИЧНОГО ТРАСУВАННЯ

О.В. Бабич

Ставиться і вирішується завдання оптимізації керування розробкою програмної системи (ПС) з метою забезпечення якості створюваного програмного продукту. Сформовано метод керування розробкою ПС, заснований на вдосконаленні технологій P-MF та зворотного семантичного трасування. Процес розробки представлено у вигляді ітеративного процесу з контролем виконання вимог до нового проекту зі створення ПС. Для оцінки наближення описів проекту застосовуються якісні експертні оцінки, які порівнюються з контрольним поданням наближення, що задається командою проектувальників. Запропонований підхід дозволяє забезпечити якість створюваного програмного продукту, знижує ризики проектування та терміни розробки ПС.

Ключові слова: *якість програмного продукту, методологія P-MF, RST, зворотне семантичне трасування, лексикографічне впорядкування варіантів, експертне оцінювання.*

MANAGEMENT OF SOFTWARE DEVELOPMENT PROCESS USING P-MODELING FRAMEWORK AND REVERSE SEMANTIC TRACEABILITY

O.V. Babich

A problem of optimizing the management of software development process was formulated and solved. Proposed method is based on the improvement of P-MF methodology and the Reverse Semantic Traceability. The development process is presented in the form of an iterative process to meet all the requirements of the new software project. To assess the proximity of the project description we use qualitative evaluation of experts, which are compared with the reference representation given by the design team. The proposed approach makes it possible to ensure the quality of the developed software, reduces the risks and duration of software development process.

Keywords: *software quality, P-MF, RST, reverse semantic traceability, lexicographical ordering, expert assessments.*