

# МЕРЫ КАЧЕСТВА СПЕЦИФИКАЦИЙ ТРЕБОВАНИЙ К ПРОГРАММНОМУ СРЕДСТВУ

А.Ю. Чиркова, В.В. Бахтизин

Кафедра программного обеспечения информационных технологий,  
Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь

E-mail: aliaksandra.chyrkova@gmail.com, bww@bsuir.by

*Предлагаются меры (метрики) качества спецификаций требований к программному средству; приводится пример, иллюстрирующий достоинства предложенных мер.*

## ВВЕДЕНИЕ

Тщательно разработанные требования к программному средству (ПС) являются ключевым моментом, влияющим на успех проекта. Стоимость исправления ошибки после завершения проекта на гораздо больше, чем расходы по ее исправлению на этапе анализа требований. Поскольку требования часто меняются, важно контролировать эти изменения, чтобы иметь возможность предвидеть и реагировать на запросы на внесение изменений.

Каждый процесс жизненного цикла ПС может служить механизмом улучшения его качества. Существует несколько подходов совершенствования этих процессов: «Модель зрелости возможностей создания ПО» (SW-CMM) [1], «Цель, Вопрос, Метрика» (GQM) [2]. Эти подходы дополняют друг друга, но они редко применяются вместе. Подходы, основанные на оценке, всегда должны включать в себя измерения, однако, SW-CMM помогая организациям, разрабатывающим программное обеспечение, улучшить зрелость своих процессов, не позволяет оценивать их. Поэтому в данной работе описывается один из основных ключевых процессов в SW-CMM – управление требованиями.

## I. УПРАВЛЕНИЕ ТРЕБОВАНИЯМИ ПО SW-CMM И ПАРАДИГМА БЕЙЗИЛИ

Управление требованиями является одним из процессов CMM. Как определено в [3], «Целью управления требованиями является создание понимания между заказчиком и всеми лицами, задействованными в разработке ПС, требований заказчика». Это означает, что требования к ПС должны быть полными, недвусмысленными, контролируруемыми, трассируемыми и т.д. для того, чтобы разработать ПС, которое удовлетворяет потребностям клиента. Управление требованиями фокусируется на контроле сбора требований, на управлении изменениями требований, и представляет собой процесс обеспечения того, что ПС будет отвечать этим требованиям.

Парадигма Бейзили (goal, question, metric, GQM) является методом, помогающим сфокусироваться на формировании целей. Парадигма GQM состоит из трех этапов: указания целей,

исходя из потребностей организации; формулирования количественных вопросов; определения набора метрик(мер).

CMM и GQM очень легко переплетаются между собой. Для каждого ключевого процесса CMM определяет цели, которые могут быть использованы для первого этапа GQM. CMM определяет две цели для процесса управления требованиями. Первая заключается в следующем: «Системные требования, предъявляемые к ПС, должны быть контролируемыми и являться основой для проектирования ПС и диспетчеризации хода выполнения проекта» [3]. Вторая цель заключается в следующем: «Планы разработки ПС, продукция и действия сохраняют непротиворечивость с предъявляемыми системными требованиями» [3].

Достижение этих целей подразумевает наличие: системы разработки технических заданий на ПС; системы заявок и уточнений и согласования требований на протяжении всего жизненного цикла ПС. Данные цели могут быть перефразированы таким образом, чтобы они вписывались в концепцию GQM.

Анализируя первую цель, вопрос возник: каким образом можно контролировать требования? Любая информация о требованиях может улучшить контроль над ними. Для сравнения важно анализировать требования на всех стадиях создания требований.

Задача второй цели – сохранить согласованность между требованиями и ПС, поэтому предлагается поддерживать трассируемость требований во всех документах, используемых при разработке ПС.

## II. МЕРЫ КАЧЕСТВА ТРЕБОВАНИЙ

Спецификация требований (СТ) к ПС является документом или базой данных, которые содержат требования. Для каждого требования предлагается включить дополнительную информацию, например: состояние требования, источник, приоритета и т.д., потому что эта информация может помочь в дальнейшей оценке стабильности требований.

Существует причинно-следственная связь между размером и стабильностью требований:

если размер технического задания меняется хаотично, то требования могут быть нестабильными. Внутренний атрибут «размер СТ» может измеряться количеством требований. Подсчет требований может производиться в разные моменты жизненного цикла ПС. Общее количество требований является итоговой суммой функциональных и нефункциональных требований. Количество исходных, текущих и окончательных требований эквивалентны общему количеству требований. Эти меры отличаются друг от друга временем и причиной подсчета требований.

Исходное количество требований – это сумма всех функциональных и нефункциональных требований заказчика. Окончательное количество требований – это сумма всех требований, реализованных в ПС. Текущее количество требований – это сумма требований, собранных в текущий момент времени.

Сравнивая данные, получаемые в процессе работы над требованиями, можно видеть, насколько общий объем потребностей меняется со временем. На основании этого можно сделать прогноз того, как размер СТ будет меняться с течением времени.

Различные состояния требований соответствуют фазам жизненного цикла СТ. В течение жизненного цикла статусы требований могут меняться. Предлагается отслеживать статусы требований с момента его создания до его поставки готового программного продукта заказчику. Эта мера позволяет иметь представление о том, как управлять, например, теми требованиями, которые еще находятся в стадии разработки, но уже нужны на определенном этапе разработки программного продукта. Отношения статусов не могут быть эмпирическими отношений, а только бинарными, и могут принимать значения «равны» и «различны». Для получения качественной оценки статусов требований предлагается использовать меру полноты СТ, которая может быть выражена следующей формулой:

$$M_{completeness} = 1 - \frac{N_{TBD}}{N_R}, \quad (1)$$

где  $N_{TBD}$  – количество требований, подлежащих определению,  $N_R$  – количество всех требований.

Изменение требования – это любая семантическая модификация требования, т.е. это либо исправление неточности или ошибки, добавление новой функции к системе, удаление или обновление функции. Предлагается отслеживать каждое изменение требования. Внутренними атрибутами этой меры изменения могут быть: размер изменения, причина изменения и др. Для оценки устойчивости СТ к изменениям следует принимать во внимание только те требования, которые были изменены или удалены, т.к. эти требования зависят от общего количества требований в спецификации, сумма измененных и удаленных требований не может быть больше общего количества требований в спецификации. Новые тре-

бования не зависят от общего количества требований. Таким образом, мера устойчивости требований к изменениям может быть оценена по следующей формуле:

$$M_{stability} = \frac{1}{n} \sum_{i=1}^n \left( 1 - \frac{R_{changed_i} + R_{removed_i}}{R_{current_i}} \right), \quad (2)$$

где  $n$  – число версий спецификации,  $R_{changed_i}$  – число измененных требований в  $i$ -ой версии,  $R_{removed_i}$  – число удаленных требований в  $i$ -ой версии,  $R_{current_i}$  – число текущих требований в  $i$ -ой версии.

Показателем качества управления требованиями может служить соответствие требований выполненным работам. В ходе работ некоторые требования могут изменяться, но изменения могут быть не внесены в СТ, некоторые требования могут быть не реализованы вообще, кроме того, некоторые функциональности могут быть разработаны, но при этом вообще не описаны. Все эти несоответствия влияют на качество готового ПС, т.к. при тестировании неучтенные изменения и требования могут быть рассмотрены как ошибки в ПС. Более того, не описанные функциональности могут противоречить описанным требованиям. В целом несоответствия влияют так же и на сопровождение ПС и т.д. Мера соответствия требований и ПС может быть выражена в следующей формуле:

$$M_{compl} = 1 - \frac{R_{ignored} + R_{incomplete} + R_{undescribed}}{R_{all} + R_{undescribed}}, \quad (3)$$

где  $R_{ignored}$  – число не реализованных требований,  $R_{incomplete}$  – число несоответствующих (т.е. отличающихся от описанных) требований,  $R_{undescribed}$  – число неописанных, но реализованных требований и функциональностей,  $R_{undescribed}$  – число всех требований, описанных в спецификации.

## ЗАКЛЮЧЕНИЕ

Предложенные меры качества СТ к ПС могут быть применены для оценки качества СТ с целью улучшения качества требований и, следовательно, качества ПС, а так же сбора статистики с целью улучшения ключевого процесса разработки ПС – управления требованиями.

1. Paulk, M. C., Curtis, B., Chrissis, M. B. and Weber, C. V. Capability Maturity Model for Software Version 1.1, Software Engineering Institute Technical Report, CMU/SEI-93-TR-24, ESC-TR-93-177, Pittsburgh, PA, 1993.
2. Basili, V. R. Rombach, H. D. The TAME project: Towards improvement-oriented software environments, in IEEE Transactions on Software Engineering 14(6), 1988, pp. 758-773.
3. Paulk, M. C., Weber, C. V., Garcia, S., Chrissis, M. B., Bush, M. Key Practices of the Capability Maturity Model Version 1.1, Software Engineering Institute Technical Report, CMU/SEI-93-TR-25 ESC-TR-93-178, Pittsburgh, PA, 1993.