

Использование метрик в процессе обеспечения качества сложных информационных систем

Гребенюк В.М.

Аннотация—В статье рассматривается метод для выбора, систематизации и анализа метрик в процессе обеспечения качества сложных информационных систем. Описываемый метод позволяет получить более точное и однозначное представление о текущем уровне риска с учётом критичности подсистем, по сравнению с использованием одиночных метрик.

Ключевые слова—обеспечение качества, метрики, сложные системы, информационные системы.

I. ВВЕДЕНИЕ

В процессе обеспечения качества сложных информационных систем часто возникает необходимость в разносторонней оценке осуществляемых мероприятий для своевременного выявления рисков и их устранения. Многообразие доступных для измерения характеристик и разнородность элементов системы делает задачу внедрения метрик труднорешаемой, ведь важно не только правильно определить необходимые критерии и методы оценки, но и разработать шкалу результатов оценки и эталонные значения.

II. СУЩЕСТВУЮЩИЕ МЕТРИКИ

На данный момент в литературе представлен широкий спектр различных метрик и показателей, применяемых для программного обеспечения и систем в целом. При этом не существует единого подхода к их использованию, как и универсального набора метрик: стандарты и лучшие практики либо ограничиваются общим сводом правил формирования или использования метрик, оставляющим свободу для их трактования; либо сводятся к списку узкоспециализированных метрик, не подлежащему применению в широком спектре инициатив.

Нет единого подхода и в классификации метрик, однако в общем случае, предлагается выделять три категории метрик по их объекту, среди которых метрики продукта (размер, сложность, производительность и др.), процесса (эффективность исправления дефектов, особенности поступления дефектов и др.) и проекта

(количество вовлеченных разработчиков, стоимость, календарный план и т.д.) [1]. В дополнение к этому, каждая из категорий может быть разбита на более мелкие подкатегории, например, метрики процесса тестирования могут быть подразделены на метрики эффективности, стоимости, времени выполнения, предсказуемости и зрелости процессов тестирования [2]. Но, и такое разделение условно: некоторые метрики всё равно могут быть отнесены к нескольким категориям [1].

С точки зрения методов формирования показателей, можно отметить «прямые» метрики, основанные на абсолютных значениях, а также производные показатели, являющиеся результатов обработки абсолютных показателей по установленным правилам [3].

Говоря о многообразии существующих метрик, многие авторы сходятся во мнении, что для определения необходимого в конкретной ситуации набора показателей рекомендуется использовать Goal/Question/Metric подход [1]. Суть данного метода состоит в том, чтобы сначала выделить конечные цели, далее задать вопросы о том, что может выступать в качестве их характеристики, а затем на основе полученных ранее данных выбрать подходящие метрики.

Например, применяя такой подход и заявляя повышение эффективности тестирования как конечную цель, в качестве характеристик поставленной цели (то есть эффективного процесса тестирования в данном случае) можно отметить:

- Находить как можно больше дефектов до выпуска продукта;
- Находить дефекты как можно раньше относительно жизненного цикла продукта;
- Находить наиболее критичные дефекты как можно раньше в процессе тестирования;
- Как можно точнее определять причину дефекта;
- Снизить количество отпарованных дефектов, позже признанных некорректными.

В приведенном выше примере, в качестве предлагаемых к применению метрик можно выделить метрики «утечки» дефектов на следующую фазу тестирования или стадию жизненного цикла продукта (defect leakage или количество дефектов, которое могло быть найдено на более ранних фазах тестирования или стадиях жизненного цикла продукта, но было найдено

Статья получена 21 марта 2014.

Гребенюк В.М. – аспирант, Московский государственный технический университет радиотехники, электроники и автоматики (e-mail: viktor.grebenyuk@gmail.com).

позже), статистика обнаружения новых дефектов по времени или фазам тестирования с учётом их критичности (new defect arrival pattern), статистика перевода дефектов по статусам, сделанным в рамках уточнения деталей дефекта, а также отношение признанных некорректными дефектов к общему количеству найденных дефектов (defect rejection rate).

Решая задачу о внедрении метрик, стоит обратить особое внимание на рекомендации некоторых авторов придерживаться минимально необходимого набора метрик. Авторы отмечают, что многообразие применяемых метрик может негативно отразиться на производительности команды – сбор и упорядочивание дополнительной информации, в последствии не приносящей дополнительной стоимости или пользы не только увеличивает объём работы, но и может оказать демотиивирующее воздействие на участников процесса [4].

Необходимо также добавить, что внедрение некоторых видов метрик следует производить с осторожностью, потому что такие метрики могут негативно влиять на качество разрабатываемых систем в силу создания неправильно ориентированной мотивации сотрудников, например, стремление скрыть информацию об известных дефектах или сознательное искажение описания дефекта с целью понижения его приоритета [5].

III. ОТ ЧАСТНОГО К ОБЩЕМУ

Проведенный анализ рассмотренных в литературе метрик, а также вариантов их использования, выявил, что показатели, основанные на абсолютных значениях количества выполненных тестов, числе найденных или исправленных дефектов, часто не могут быть использованы в чистом виде, например, без привязки к статистическим данным, моменту времени или этапу проекта - без дополнительной информации, значения таких показателей могут быть трактованы по-разному разными участниками проекта. По сравнению с абсолютными значениями, понимание тенденции, направления вектора изменений или выделение общего тренда может быть гораздо информативнее.

Для упорядочивания различных видов и типов метрик, которые могут быть потенциально использованы для обеспечения качества сложных информационных систем, предлагается выделить две основные группы метрик показателей: метрики, описывающие качество продукта и метрики, характеризующие качество процесса.

Например, в качестве метрик качества продукта могут быть выделены [1]:

- Время наработки на отказ;
- Распределение дефектов относительно количества строк кода продукта или его функциональных областей;
- Распределение проблем, отрапортованных пользователями продукта, по видам;
- Уровень удовлетворённости работой продукта конечными пользователями.

В качестве примера метрик процесса можно привести:

- статистика исправления дефектов по фазам;
- эффективность исправления дефектов;
- распределение дефектов по фазам тестирования;
- статистика нахождения дефектов во время тестирования.

Не смотря на то, что данные группы метрик связаны (например, чем выше степень покрытия системы тестами, чем больше вероятность обнаружения скрытых дефектов), метрики качества процессов можно однозначно отделить от метрик качества продукта (т.е. степень покрытия тестами - не то же самое, что высокое качество тестируемой системы) [6].

IV. ОСОБЕННОСТИ СЛОЖНЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

Использование метрик в процессе обеспечения качества сложных информационных систем имеет свои особенности, связанные как с особенностями строения таких систем, так и с особенностями самого процесса обеспечения качества для таких систем.

Виду таких особенностей, при использовании метрик в рамках обеспечения качества сложных информационных систем необходимо учесть следующие ограничения:

- Наличие нескольких подсистем разной степени критичности, часто основанных на разных технологиях и языках программирования.
- Возможное отсутствие доступа к исходному коду продукта для отдельных подсистем (отдельная подсистема может быть построена на основе «коробочного» решения или решения, разработанного сторонним поставщиком, например).
- Качество процессов аналитики, разработки, тестирования и др. может отличаться между подсистемами, а значит результаты некоторых метрик нельзя аппроксимировать с одной подсистемы на другие подсистемы (например, это касается показателя «плотность дефектов»).
- Может отсутствовать унификация между информацией о найденных дефектах, деталях выполняемых испытаний и др., поступающей от разных подсистем. Кроме того, такая информация может отличаться по составу (полноте).

Таким образом, можно сделать вывод о необходимости упорядочивания метрик процесса и продукта для системы в целом, заодно с выделением уровня критичности подсистем.

V. ОПРЕДЕЛЕНИЕ УРОВНЯ КРИТИЧНОСТИ ПОДСИСТЕМ

Из-за того, что в состав сложных информационных систем могут входить несколько подсистем разной степени критичности, то метрики и показатели, получаемые от этих подсистем, также должны восприниматься с отличным друг от друга и однозначно определённым уровнем критичности или приоритетности в отношении всей системы.

В качестве аналогии можно привести пример города: если в одном из районов отключается электричество, то это может не оказывать существенного влияния на

функционирование остальной части города, при условии что отключение электричества не длится слишком долго, не затрагивает какие-либо критичные объекты или повторяется слишком часто. Из-за того, что город, как и многие сложные информационные системы, предоставляет различные распределённые виды сервисов разным потребителям, оценка общей успешности предоставления сервисов городом будет частично зависеть от видов запросов и соответствующих ожиданий разных групп потребителей, и частично от качественных характеристик предоставленных сервисов. Понимание таких критериев успешности предоставления сервисов системой является важным аспектом системы мониторинга и оценки сложных систем [7].

Для формирования градации критичности подсистем и соответствующих метрик, предлагается использовать принципы, описанные в ГОСТ Р 51901-2002 "Управление надежностью. Анализ риска технологических систем" [8] и ГОСТ 27.310-95 "Надежность в технике. Анализ видов, последствий и критичности отказов. Основные положения" [9] для определения перечня критичных элементов. В частности, согласно упомянутому стандарту ГОСТ 27.310-95, в перечни критичных элементов включают:

- элементы, возможная тяжесть последствий отказов которых, оцениваемая качественно или количественно, превосходит допустимый для рассматриваемого объекта уровень;
- элементы, отказы которых неизбежно вызывают полный отказ объекта;
- элементы с ограниченным сроком службы (ресурсом), не обеспечивающем требуемой долговечности объекта;
- элементы, по которым в момент проведения анализа отсутствуют достоверные данные об их качестве и надежности в рассматриваемых условиях применения и/или возможных последствиях их отказов.

Таким образом, в список наиболее критичных подсистем попадут подсистемы, отвечающие следующим критериям:

- подсистемы, используемые в наибольшем количестве бизнес процессов;
- подсистемы, используемые в наиболее важных бизнес процессах (в том числе, являются определяющими успешность выполнения основных бизнес процессов другими подсистемами);
- подсистемы, изначально имеющие дополнительные риски (например, график поставки изначально спланирован слишком оптимистично);
- подсистемы, по которым в момент проведения анализа отсутствуют достоверные данные об их

критичности для системы в целом.

VI. ПРЕДЛАГАЕМОЕ РЕШЕНИЕ ДЛЯ СИСТЕМАТИЗАЦИИ И АНАЛИЗА МЕТРИК КАЧЕСТВА СЛОЖНЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

Конечная цель любых метрик – своевременная идентификация рисков. В случае обеспечения качества сложных систем, как и во многих других случаях, основными рисками являются:

- Риски не завершить запланированные активности в срок (т.е. отклонение от календарного плана);
- Риски возникновения новых, незапланированных или неучтённых ранее активностей в процессе работы над инициативой (т.е. отклонение по составу работ и/или по трудозатратам);
- Риски возникновения доработок и/или исправления проблем после окончания инициативы (например, выявление необходимости доработок или исправлений после завершения проекта, если речь идёт о проекте).

В общем случае, при использовании метрик в сложных информационных системах, необходимо получать как можно более точное и однозначное представление о текущем уровне риска с учётом критичности подсистем.

Для решения данной задачи предлагается обобщить данные, полученные с помощью отдельных метрик, в общие индикативные показатели уровня качества продукта и качества процессов для каждой из подсистем. Для этого предлагается использовать шкалу оценки от 0 до 10 (чем выше качество, тем выше оценка).

Данные индикативных показателей уровня качества продукта и качества процессов для каждой из подсистем предлагается свести в общую таблицу, дополнив её информацией об уровне критичности каждой из подсистем, выставленных по шкале от 1 до 5 (тем выше риск, тем выше оценка) на основе критериев определения критичности подсистем, рассмотренных ранее.

Дополнительно, рекомендуется также указать информацию о наиболее важных зависимостях одной подсистемы от другой.

Пример полученной сводной информации о подсистемах и соответствующих значениях метрик приведён в Таблице 1.

На основе полученной таблицы предлагается построить пузырьковую диаграмму (bubble graph), где уровень критичности каждой подсистемы влияет на размер «пузырька», отображающего соответствующую систему.

Таблица 1. Пример сводной информации о подсистемах и соответствующих значениях метрик.

| System ID | System Name | Product Quality | Process Quality | Criticality | Depends on (System ID) |
|-----------|-------------|-----------------|-----------------|-------------|------------------------|
| 1 | System A | 2,46 | 1,75 | 1,00 | 2 |
| 2 | System B | 3,58 | 6,78 | 2,00 | |
| 3 | System C | 7,78 | 6,45 | 1,00 | 4 |
| 4 | System D | 5,30 | 7,02 | 4,00 | 5 |
| 5 | System E | 4,80 | 3,40 | 5,00 | |

Данная мера позволяет нагляднее определить «вес» системы на диаграмме, относительно других систем. Пример такой диаграммы, построенной на основе данных Таблицы 1, приведен на Рисунке 1.

Диаграмма, изображенная на Рисунке 1 разбита на четверти с использованием принципов, схожих с теми, что описаны в [1]. Каждая из четвертей, указанных на диаграмме, характеризует изображенные на ней подсистемы следующим образом:

- I четверть (высокое качество процессов и продукта) – оптимальное соотношение уровня качества процессов и продукта (пример – большое количество активностей по обеспечению качества, выполненное в срок, позволило обнаружить малое количество дефектов в работе системы);

- II четверть (высокое качество процессов, но низкое качество продукта) – хорошее/неплохое состояние (пример – большое количество активностей по обеспечению качества, выполненное в срок, позволило обнаружить большое количество дефектов в работе системы). Не смотря на низкое качество продукта, высокое качество процессов может обеспечить рост качества продукта до заданных величин.

- III четверть (низкое качество процессов и продукта) – наихудшее состояние (пример – малое количество активностей по обеспечению качества, выполненное с опозданием, позволило обнаружить большое количество дефектов в работе системы). Низкое качество процессов в совокупности с низким качеством продукта говорит о наивысшем уровне риска.

- IV четверть (низкое качество процессов, но высокое качество продукта) – неопределённое состояние (пример – малое количество активностей по обеспечению качества, выполненное с опозданием, позволило обнаружить небольшое количество дефектов в работе системы). Качество процессов слишком низко для того, чтобы идентифицировать объективный уровень качества продукта.

Дополнительно, на диаграмму нанесено отображение зависимостей подсистем (рекомендуется использовать правила отображения зависимостей, сходные с теми, что используются для UML-схем [10]). Это сделано для того, чтобы предоставить возможность получения дополнительной информации о рисках, связанных с зависимыми системами. Например, согласно информации, изображенной на Рисунке 1, существует

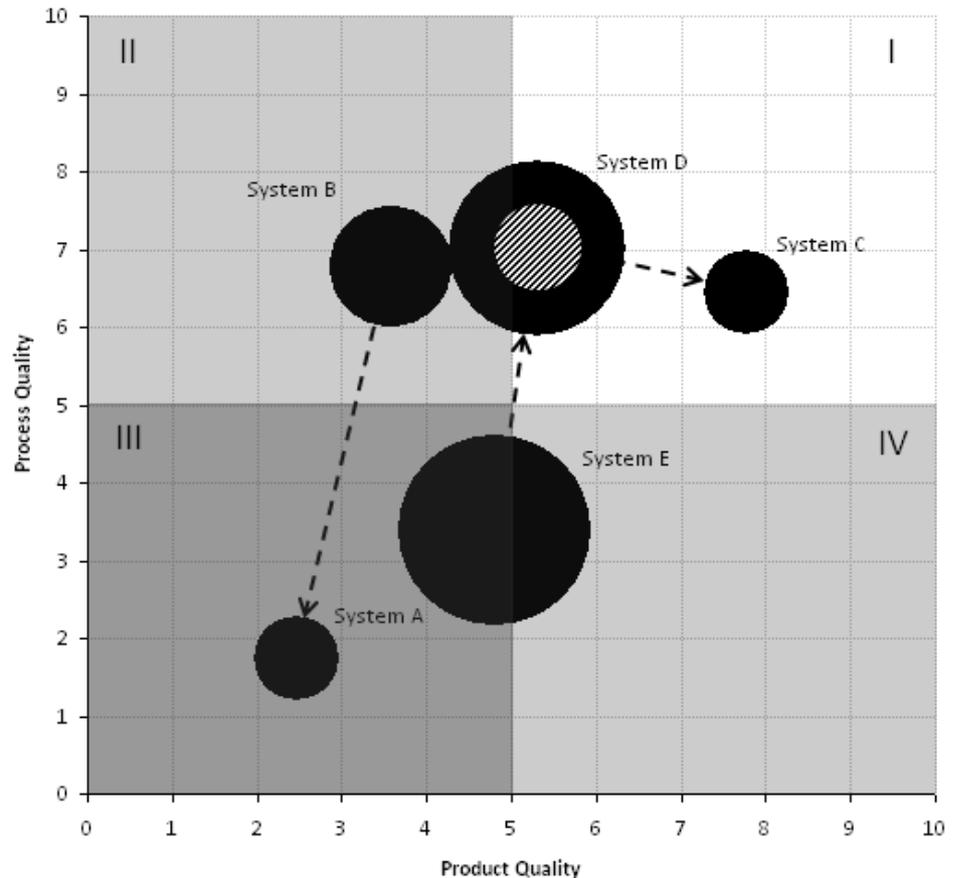


Рисунок 1. Пример пузырьковой диаграммы, построенной на основе данных Таблицы 1.

зависимость подсистемы “А” от подсистемы “В”. Это, в частности, означает, что низкое качество подсистемы “В” создаёт дополнительные риски для работоспособности подсистемы “А”.

Кроме того, в некоторых случаях возможно изменение оценки критичности отдельных систем – в таких случаях предлагается отображать на диаграмме и текущее и предыдущее значение оценки критичности (например, отображать предыдущее значение оценки критичности подсистемы с использованием другого типа заливки, как это показано на примере подсистемы “D”, изображенной на Рисунке 1).

VII. ЗАКЛЮЧЕНИЕ

Предлагаемое решение для систематизации и анализа метрик качества в сложных информационных системах, позволяет получить более точное и однозначное представление о текущем уровне риска с учётом критичности подсистем, по сравнению с использованием одиночных метрик. Наглядное отображение позволяет отказаться от необходимости обязательного детального пояснения результатов сбора метрик во избежание неправильного восприятия полученных показателей.

БИБЛИОГРАФИЯ

- [1] Kan S.H., Metrics and Models in Software Quality Engineering, Second Edition. - Addison Wesley, 2002. - 560 с.
- [2] Certified Tester, Expert Level Syllabus, Ed. by Veenendaal E., Bath G., Evans I. - ISTQB, 2011 Режим доступа: <http://www.istqb.org/downloads/finish/18/12.html> (дата обращения 19.03.2014)

- [3] Kaner C., Bond W.P., Software Engineering Metrics: What Do They Measure and How Do We Know?, 10th International Software Metrics Symposium, Metrics 2004. Режим доступа: <http://testingeducation.org/a/3> (дата обращения 19.03.2014)
- [4] Hoffman D. The Darker Side of Metrics // Сайт компании Software Quality Methods, LLC. Режим доступа: <http://www.softwarequalitymethods.com/Papers/DarkMets%20Paper.pdf> (дата обращения 19.03.2014)
- [5] Гребенюк В.М. О методах определения эффективности и достаточности мер по обеспечению качества сложных информационных систем // INTERMATIC – 2013, Материалы Международной научно-технической конференции, 2 – 6 декабря 2013 г., Москва, часть 6, с. 11-12.
- [6] SWEBOOK v3.0, edited by Bourque P., Fairley R.E. – IEEE, 2014 Режим доступа: www.swebok.org (дата обращения 19.03.2014)
- [7] Northrop L. et al, Ultra-Large-Scale Systems: The Software Challenge of the Future. - Carnegie Mellon University, 2006. - 150 с.
- [8] ГОСТ Р 51901-2002 "Управление надежностью. Анализ риска технологических систем"
- [9] ГОСТ 27.310-95 "Надежность в технике. Анализ видов, последствий и критичности отказов. Основные положения"
- [10] Схемы компонентов UML: справочные материалы // Microsoft Developer Network. Режим доступа: [http://msdn.microsoft.com/ru-ru/library/dd409390\(v=vs.100\).aspx](http://msdn.microsoft.com/ru-ru/library/dd409390(v=vs.100).aspx) (дата обращения 19.03.2014)

Using Quality Assurance Metrics for Complex Information Systems

Grebenyuk V.M.

Abstract – The article describes the method to select, classify and analyze Quality Assurance metrics for Complex Information Systems. The proposed method allows getting representation of the current risk level along with specifying criticality of subsystems more precisely in comparing with using a set of various metrics separately from each other.

Keywords – quality assurance, metrics, complex systems, information systems.