

# **Качество программного обеспечения**

Переход от штучной разработки программных продуктов к промышленному программированию обусловил повышение требований к качеству создаваемого ПО. В настоящее время существует несколько стандартов, связанных с оценкой качества этих процессов, которое обеспечивает компания-разработчик. К наиболее интересным для рассмотрения относятся:

- **международные стандарты** серии ISO 9000 (ISO 9000 — ISO 9004);
- **CMM — Capability Maturity Model — модель зрелости (совершенствования) процессов создания программного обеспечения**, предложенная SEI (Software Engineering Institute — институт программирования при университете Карнеги — Меллон);
- **процесс сертификации программ** на базе информации об их использовании.

# Серия стандартов ISO 9000

Одной из важнейших проблем обеспечения качества программных средств является **формализация характеристик качества и методология их оценки**. Для определения адекватности качества функционирования, наличия технических возможностей программных средств к взаимодействию, совершенствованию и развитию необходимо использовать стандарты в области оценки характеристик их качества. Основой регламентирования показателей качества программных средств ранее являлся международный стандарт ISO 9126:1991 (ГОСТ Р ИСО/МЭК 9126-93) «Информационная технология. Оценка программного продукта. Характеристики качества и руководство по их применению». Завершается разработка и формализован последний проект состоящего из четырех частей стандарта ISO 9126-1—ISO 9126-4 для

# Серия стандартов ISO 9000

Проект состоит из следующих частей под общим заголовком «Информационная технология — характеристики и метрики качества программного обеспечения»:

«Часть 1. Характеристики и субхарактеристики качества;

«Часть 2. Внешние метрики качества»;

«Часть 3. Внутренние метрики качества»;

«Часть 4. Метрики качества в использовании»

# Серия стандартов ISO 9000

В России в области обеспечения жизненного цикла и качества сложных комплексов программ в основном применяется группа устаревших ГОСТов, которые отстают от мирового уровня на 5—10 лет.

**Первая часть стандарта — ISO 9126-1** — распределяет атрибуты качества программных средств по шести характеристикам, используемым в остальных частях стандарта. Исходя из принципиальных возможностей их измерения все характеристики могут быть объединены в три группы, к которым применимы разные категории метрик:

- **категорийным**, или **описательным** (номинальным), метрикам наиболее адекватны **функциональные возможности программных средств**;
- **количественные метрики** применимы **для измерения надежности и эффективности** сложных комплексов программ;
- **качественные метрики** в наибольшей степени соответствуют **практичности, сопровождаемости и мобильности** программных средств.

# Серия стандартов ISO 9000

- **Вторая и третья части стандарта** — ISO 9126-2 и ISO 9126-3 — посвящены **формализации соответственно внешних и внутренних метрик характеристик качества** сложных программных средств. Все таблицы содержат унифицированную рубрикацию, где отражены имя и назначение метрики; метод ее применения; способ измерения, тип шкалы метрики; тип измеряемой величины; исходные данные для измерения и сравнения; а также этапы жизненного цикла программного средства (по ISO 12207), к которым применима метрика.
- **Четвертая часть стандарта** — ISO 9126-4 — предназначена **для покупателей, поставщиков, разработчиков, сопровождающих, пользователей и менеджеров качества программных средств**. В ней обосновываются и комментируются выделенные показатели сферы (контекста) использования программных средств и группы выбранных метрик для пользователей.

# Выбор показателей качества

**Исходными данными и высшим приоритетом** при выборе показателей качества в большинстве случаев являются **назначение, функции и функциональная пригодность соответствующего программного средства**. Достаточно полное и корректное описание этих свойств должно служить базой для определения значений большинства остальных характеристик и атрибутов качества. Принципиальные и технические возможности и точность измерения значений атрибутов характеристик качества всегда ограничены в соответствии с их содержанием. Это определяет рациональные диапазоны значений каждого атрибута, которые могут быть выбраны на основе здравого смысла, а также путем анализа прецедентов в спецификациях требований реальных проектов.

# Выбор показателей качества

Процессы выбора и установления метрик и шкал для описания характеристик качества программных средств можно разделить на два этапа:

- **выбор и обоснование набора исходных данных, отражающих общие особенности и этапы жизненного цикла проекта программного средства и его потребителей, каждый из которых влияет на определенные характеристики качества комплекса программ;**
- **выбор, установление и утверждение конкретных метрик и шкал измерения характеристик и атрибутов качества проекта** для их последующей оценки и сопоставления с требованиями спецификаций в процессе квалификационных испытаний или сертификации на определенных этапах жизненного цикла программного средства.

# Оценка качества

Методологии и стандартизации оценки характеристик качества готовых программных средств и их компонентов (программного продукта) на различных этапах жизненного цикла **посвящен международный стандарт ISO 14598**, состоящий из шести частей. Рекомендуется следующая **общая схема процессов оценки характеристик качества программ**:

- **установка исходных требований для оценки** — определение целей испытаний, идентификация типа метрик программного средства, выделение адекватных показателей и требуемых значений атрибутов качества;
- **селекция метрик качества**, установление рейтингов и уровней приоритета метрик субхарактеристик и атрибутов, выделение критериев для проведения экспертиз и измерений;
- **планирование и проектирование** процессов оценки характеристик и атрибутов качества в жизненном цикле программного средства;
- **выполнение измерений для оценки**, сравнение результатов с критериями и требованиями, обобщение и оценка результатов.

# Оценка качества

Для каждой характеристики качества рекомендуется формировать **меры и шкалу измерений** с выделением требуемых, допустимых и неудовлетворительных значений. Реализация процессов оценки должна коррелировать с этапами жизненного цикла конкретного проекта программного средства в соответствии с применяемой, адаптированной версией стандарта ISO 12207.

- **Функциональная пригодность** — наиболее неопределенная и объективно трудно оцениваемая субхарактеристика программного средства. Области применения, номенклатура и функции комплексов программ охватывают столь разнообразные сферы деятельности человека, что невозможно выделить и унифицировать небольшое число атрибутов для оценки и сравнения этой субхарактеристики в различных комплексах программ.

# Оценка качества

- **Оценка корректности программных средств** состоит в формальном определении степени соответствия комплекса реализованных программ исходным требованиям контракта, технического задания и спецификаций на программное средство и его компоненты. Путем верификации должно быть определено соответствие исходным требованиям всей совокупности компонентов комплекса программ, вплоть до модулей и текстов программ и описаний данных.
- **Оценка способности к взаимодействию** состоит в определении качества совместной работы компонентов программных средств и баз данных с другими прикладными системами и компонентами на различных вычислительных платформах, а также взаимодействия с пользователями в стиле, удобном для перехода от одной вычислительной системы к другой с подобными функциями.

# Оценка качества

- **Оценка защищенности программных средств** включает определение полноты использования доступных методов и средств защиты программного средства от потенциальных угроз и достигнутой при этом безопасности функционирования информационной системы. Наиболее широко и детально методологические и системные задачи оценки комплексной защиты информационных систем изложены в трех частях стандарта ISO 15408:1999-1— ISO 15408:1999-3 «Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий».
- **Оценка надежности** — измерение количественных метрик атрибутов субхарактеристик в использовании: завершенности, устойчивости к дефектам, восстанавливаемости и доступности/готовности.

# Оценка качества

- **Потребность в ресурсах памяти и производительности** компьютера в процессе решения задач значительно изменяется в зависимости от состава и объема исходных данных. Для корректного определения предельной пропускной способности информационной системы с данным программным средством нужно измерить экстремальные и средние значения длительностей исполнения функциональных групп программ и маршруты, на которых они достигаются. Если предварительно в процессе проектирования производительность компьютера не оценивалась, то, скорее всего, понадобится большая доработка или даже замена компьютера на более быстродействующий.
- **Оценка практичности** программных средств проводится экспертами и включает определение понятности, простоты использования, изучаемости и привлекательности программного средства. В основном это качественная (и субъективная) оценка в баллах, однако некоторые атрибуты можно оценить количественно по трудоемкости и длительности выполнения операций при использовании программного средства, а также по объему документации, необходимой для их изучения.

# Оценка качества

- **Сопровождаемость** можно оценивать полнотой и достоверностью документации о состояниях программного средства и его компонентов, всех предполагаемых и выполненных изменениях, позволяющей установить текущее состояние версий программ в любой момент времени и историю их развития. Она должна определять стратегию, стандарты, процедуры, распределение ресурсов и планы создания, изменения и применения документов на программы и данные.
- **Оценка мобильности** — качественное определение экспертами адаптируемости, простоты установки, совместимости и замещаемости программ, выражаемое в баллах. Количественно эту характеристику программного средства и совокупность ее атрибутов можно (и целесообразно) оценить в экономических показателях: стоимости, трудоемкости и длительности реализации процедур переноса на иные платформы определенной совокупности программ и данных.

# Система управления качеством

**Выбор характеристик и оценка качества программных средств** — лишь одна из задач в области обеспечения качества продукции, выпускаемой компаниями разработчиками ПО.

**Комплексное решение задач обеспечения качества программных средств** предполагает разработку и внедрение той или иной **системы управления качеством**. В мировой практике наибольшее распространение получила **система, основанная на международных стандартах серии ISO 9000**, включающей десяток с лишним документов, в том числе стандарт, регламентирующий обеспечение качества ПО (ISO 9000/3). Эти стандарты должны служить **руководством для ведущих специалистов компаний, разрабатывающих ПО на заказ**.

# Определения характеристик и субхарактеристик качества (ISO 9126-1):

- **Функциональные возможности** — способность программного средства обеспечивать решение задач, удовлетворяющих сформулированные потребности заказчиков и пользователей при применении комплекса программ в заданных условиях.
- **Функциональная пригодность** — набор и описания субхарактеристики и ее атрибутов, определяющие назначение, номенклатуру, основные, необходимые и достаточные функции программного средства, соответствующие техническому заданию и спецификациям требований заказчика или потенциального пользователя.

# Определения характеристик и субхарактеристик качества (ISO 9126-1):

- **Правильность (корректность)** — способность программного средства обеспечивать правильные или приемлемые для пользователя результаты и внешние эффекты.
- **Способность к взаимодействию** — свойство программных средств и их компонентов взаимодействовать с одной или большим числом компонентов внутренней и внешней среды.
- **Защищенность** — способность компонентов программного средства защищать программы и информацию от любых негативных воздействий.
- **Надежность** — обеспечение комплексом программ достаточно низкой вероятности отказа в процессе функционирования программного средства в реальном времени.

# Определения характеристик и субхарактеристик качества (ISO 9126-1):

- **Эффективность** — свойства программного средства, обеспечивающие требуемую производительность решения функциональных задач, с учетом количества используемых вычислительных ресурсов в установленных условиях.
- **Практичность (применимость)** — свойства программного средства, обуславливающие сложность его понимания, изучения и использования, а также привлекательность для квалифицированных пользователей при применении в указанных условиях.
- **Сопровождаемость** — приспособленность программного средства к модификации и изменению конфигурации и функций.
- **Мобильность** — подготовленность программного средства к переносу из одной аппаратно-

## **СММ— Capability Maturity Model — модель зрелости (совершенствования) процессов создания программного обеспечения**

В ноябре 1986 г. американский институт Software Engineering Institute (SEI) совместно с Mitre Corporation начал разработку обзора зрелости процессов разработки программного обеспечения, который был предназначен для помощи в улучшении их внутренних процессов [31].

Разработка такого обзора была вызвана запросом американского федерального правительства на предоставление метода оценки субподрядчиков для разработки ПО. Реальная же проблема состояла в неспособности управлять большими проектами. Во многих компаниях проекты выполнялись со значительным опозданием и с превышением запланированного бюджета. Необходимо было найти решение данной проблемы.

# **СММ— Capability Maturity Model — модель зрелости (совершенствования) процессов создания программного обеспечения**

В результате был выпущен стандарт СММ, Version 1.1, который до настоящего времени активно используется во всем мире.

Причины такого интереса к СММ понятны. Несмотря на то что и сами разработчики ПО, и их руководство зачастую очень хорошо знают свои постоянные проблемы, они не могут прийти к единому мнению о том, какие изменения необходимы компании в первую очередь. Без выработки единой стратегии проведения улучшений руководство не может найти взаимопонимания со своими сотрудниками относительно наиболее приоритетных задач по улучшению. Для достижения максимального результата от усилий, потраченных на улучшение процессов, необходимо иметь поэтапную стратегию развития, которая позволит улучшать зрелость процессов разработки постепенно, эволюционным путем.

# СММ— Capability Maturity Model — модель зрелости (совершенствования) процессов создания программного обеспечения

Постоянное улучшение процессов базируется на постепенном возвращении культуры компании, а не на проведении революционных инноваций. В СММ представлена схема такого постепенного улучшения, разделенная по пяти уровням зрелости процессов. Эти пять уровней представляют собой шкалу (рис. 1) для оценки уровня зрелости процессов разработки ПО в компании и для измерения их



Рис.1 Принцип последовательного повышения уровня зрелости: возможности развития организации

# СММ— Capability Maturity Model — модель зрелости (совершенствования) процессов создания программного обеспечения

## Основные характеристики каждого уровня:

1. **Начальный уровень.** Процесс разработки носит хаотический характер. Определены лишь немногие из процессов, и успех проектов зависит от конкретных исполнителей.
2. **Повторяемость.** Установлены основные процессы управления проектами: отслеживание затрат, графика работ и функциональности. Упорядочены некоторые процессы, необходимые для того, чтобы повторить предыдущие достижения на аналогичных проектах (проектах с аналогичными приложениями).
3. **Разработка.** Процессы разработки ПО и управления проектами описаны и внедрены в единую систему процессов компании. Во всех проектах используется стандартный для организации процесс разработки и поддержки ПО, адаптированный под конкретный проект.
4. **Контроль.** Собираются детальные количественные данные по функционированию процессов разработки и качеству конечного продукта. Анализируется значение и динамика этих данных.
5. **Улучшение качества.** Постоянное улучшение процессов основывается на количественных данных по процессам и на пробном внедрении новых идей и технологий.

# Процесс сертификации программ на базе информации об их использовании

Это новый подход к сертификации, который, по утверждению его разработчиков, дает надежные гарантии качества для коммерческих программных пакетов .

Методов сертификации качества программного обеспечения становится все больше и больше. Популярные подходы, основанные на процессах, такие как ISO 9000 и SEI-CMM, вынуждают создателей программного обеспечения жестко придерживаться выбранных стандартов и процессов разработки. Такие подходы зачастую требуют участия аудиторов, которые проверяют документацию производителя и то, как он выполняет данное им обещание. Но даже если аудитор по сертификации может убедиться в чистоте намерений производителя, одна эта проверка вовсе не гарантирует, что созданное программное обеспечение будет высокого качества.

# Процесс сертификации программ на базе информации об их использовании

Принимая во внимание эту проблему, можно предложить **методологию сертификации**, которая не требует, чтобы производитель давал клятвы, а аудиторы проверяли, насколько эти клятвы выполняются. Можно с уверенностью сказать, что только **полностью независимая сертификация продукта** и есть тот подход, которому могут доверять потребители.

Пригласив **третью сторону для выдачи сертификатов качества на программное обеспечение**, разработчики снимут с себя ответственность, как это делает врач, приглашая для консультации другого доктора или проводя еще один анализ. **Конечные пользователи тоже выиграют** от того, что беспристрастная оценка продукту будет даваться независимыми организациями. Таким образом, есть разумные причины для **появления учреждений, которые будут сертифицировать качество программного обеспечения**.

Организации, выполняющие такую сертификацию, называются **лабораториями по сертификации программного обеспечения** (Software Certification Laboratories — SCL). Их достоинство в том, что они смогут предоставить равные возможности всем производителям, если, конечно, каждый продукт будет тестироваться в равных условиях.

## Процесс сертификации программ на базе информации об их использовании

SCL до сих пор не получили широкого распространения именно из-за ответственности, которую берет на себя организация, выдающая сертификат. Если **сертифицированное ПО не оправдывает ожиданий при использовании** его в реальном производственном процессе, **удостоверитель, выдавший сертификат, несет за это определенную ответственность**. Вот и SCL могут потерять свое реноме и лишиться права заниматься сертификацией в случае ошибки. Чтобы несколько снизить этот риск, необходимо использовать **точные методы при принятии решений о сертификации — в идеале автоматизированные**.

## Процесс сертификации программ на базе информации об их использовании

Таким образом, SCL сталкивается с **проблемой точного определения того, как будет вести себя программная система** — самое главное, что требуется от уполномоченных по выдаче сертификатов.

**Процесс сертификации**, о котором идет речь, использует **автоматизированные процедуры**, что позволяет отказаться от услуг аудиторов. Этот процесс использует ресурсы тестирования конечными пользователями, полагаясь на доказавшие свою состоятельность методы. Предлагаемый процесс, полностью основанный на продуктах, как таковых, **оценивает качество работы программного обеспечения, а не качество процессов создания кода.**

# Сертификация ПО с участием пользователей

Если SCL и разработчики не могут выполнять адекватное тестирование продукта, то кто же может? Для решения этой задачи можно **объединить пользователей**, только надо сделать так, что пользователям это будет выгодно. После этого встает вопрос, как лучше всего использовать их **возможности сделать реальными гарантии на программное обеспечение**.

**Сертификация продуктов с помощью самих пользователей** — это наилучший способ предоставить гарантии на ПО, поскольку при этом продукт оценивается в ситуациях, когда он применяется в условиях реального мира.

Такой подход предполагает также сбор информации о характере применения продукта. Можно измерить частоту, с которой вызываются определенные функции, и просто собирать файлы ввода, которые используются для создания профилей. Эти наборы информации будут служить в качестве основы для выпуска ограниченных гарантий на ПО.

# Сертификация ПО с участием пользователей

Производитель программного обеспечения представляет конечный продукт для включения в него **резидентного инструментария тестирования**, в результате этого создается специальная, «инструментальная» копия. Производитель предоставляет копии SCL, которая передает инструментальную копию предварительно отобраным пользователям, работающим в разных отраслях. Эти **тестировщики** будут использовать продукт по согласованию с SCL. **Предлагаемая Microsoft модель бета-тестирования** — прекрасный пример того, как среди всех желающих отобрать только тех, кто действительно сможет предоставить самый большой объем полезной информации.