

## **Лекция 1 Качество программного обеспечения как предмет стандартизации согласно стандартам ГОСТ 2844-94, ISO/IEC 12207, ISO 9126 [1-4].**

Качество ПО - предмет стандартизации.

Стандарт ГОСТ 2844-94 дает определение качества ПО как совокупность свойств (показателей качества) ПО, которые обеспечивают его способность удовлетворять потребности заказчика в соответствии с назначением. Этот стандарт регламентирует базовую модель качества и показатели, главным среди них - надежность.

Стандарт ISO/IEC 12207 определил не только основные процессы ЖЦ разработки ПС, но и организационные и дополнительные процессы, которые регламентируют инженериию, планирования и управления качеством ПС.

Согласно стандарту на этапах ЖЦ должен проводиться контроль качества ПО: проверка соответствия требований проектируемому продукту и критериев их достижения; верификация и аттестация (валидация) промежуточных результатов ПО на этапах ЖЦ и измерение степени удовлетворения достигаемых отдельных показателей; тестирование готовой ПС, сбор данных об отказах, дефектах и других ошибках, обнаруженных в системе; подбор моделей надежности для оценивания надежности по полученным результатам тестирования (дефекты, отказы и др.); оценка показателей качества, заданных в требованиях на разработку ПС.

Можно определить качество ПО, как его пригодность и удобство для решения тех задач, для которых оно создано (так же, как и качество любого инструмента).

Однако у программных систем есть две особенности, отличающие их, если не от всех, то от многих других инструментов, используемых человеком в своей деятельности.

Во-первых, цели создания программной системы чаще всего сложны и включают множество аспектов. Программное обеспечение, за редким исключением, не разрабатывается для решения ровно одной задачи. Гораздо чаще это целый набор связанных задач из некоторой области. Кроме того, в него входят и экономическая эффективность использования данной системы и удобство работы с ней для того персонала, который имеется у организации-заказчика.

Во-вторых, очень часто программы используются для решения несколько не тех задач, для которых они предназначались при создании. Набор целей, для достижения которых применяется данная система, изменяется со временем, что отражается и в постоянном добавлении новых возможностей и функций в новые версии программ. Поэтому целостное и четкое понятие качества ПО определить очень нелегко. Вместо этого используют различные модели качества, систематизирующие набор аспектов, характеристик и метрик качества, рассмотрение которых необходимо для адекватной оценки качества разнообразных программ. Такие модели могут изменяться со временем, поскольку изменяются потребности индустрии производства ПО, появляются новые группы возможностей или внимание разработчиков привлекается к новым аспектам качества, ранее считавшимся несущественными. Наиболее популярной являлась модель качества ПО, зафиксированная в наборе стандартов ISO 9126 [1-4]. В несколько упрощенном виде (при рассмотрении так называемого внутреннего качества) эта модель определяет 6 основных характеристик качества программного обеспечения. Каждая характеристика уточняется при помощи некоторого набора более детальных атрибутов. Характеристики и атрибуты качества ПО по ISO 9126.

- **Функциональность.** Эта характеристика обозначает способность ПО решать определенный круг задач. Функциональность определяет, что именно делает данная программа. Атрибуты функциональности следующие: функциональная пригодность — способность решать нужный набор задач; точность выдаваемых результатов; защищенность — способность предотвращать доступ к функциям и данным ПО людям или другим системам, у которых нет прав на это; способность к взаимодействию с другими системами; и др.

- **Надежность.** Это способность ПО поддерживать определенный уровень работоспособности в заданных условиях. Надежность является вероятностной характеристикой работоспособности ПО. Атрибуты ее таковы: зрелость — обратная величина к частоте отказов ПО; устойчивость к отказам, способность выполнять определенные задачи и придерживаться некоторых ограничений даже в случае отказов и сбоев; способность к восстановлению после отказов и среднее время такого восстановления; и др.

- **Удобство использования (применения).** Удобство использования показывает, насколько ПО привлекательно, удобно в обучении работе с ним и при выполнении самой работы. К атрибутам удобства использования относятся: понятность — показатель, обратный к усилиям, затрачиваемым пользователями на понимание основных понятий и способов работы ПО и их применимости для решения нужных им задач; удобство обучения, обратное к усилиям на обучение работе с системой; удобство работы, обратное к усилиям на выполнение определенного круга задач; привлекательность, способность привлекать новых пользователей; и др.

- **Производительность (эффективность).** Это способность ПО обеспечивать необходимую работоспособность по отношению к выделяемым для этого ресурсам. В соответствии с затратами ресурсов разного вида — времени, памяти, пропускной способности сетевых соединений — выделяются и различные атрибуты производительности.

- **Переносимость.** Эта характеристика показывает сохранение работоспособности ПО при изменении его окружения. Ее атрибутами являются, например, возможность развертывания или установки ПО в различных окружениях и его адаптируемость — способность приспосабливаться к работе в различных окружениях при помощи действий, зафиксированных в документации.

- **Удобство сопровождения (сопровождаемость).** Удобство сопровождения определяет трудоемкость анализа, исправления ошибок и внесения изменений в ПО. Его атрибутами являются, в частности, удобство проведения тестирования, удобство внесения изменений и риск возникновения неожиданных эффектов при изменениях. В 2011 году принят стандарт ISO 25010 [5], заменяющий ISO 9126-1 и несколько изменяющий набор характеристик и атрибутов внутреннего качества ПО. В его рамках имеются следующие характеристики.

- **Функциональность** (теперь называемая *functional suitability*). **Функциональная пригодность** (*functional appropriateness*) — способность ПО решать нужные пользователям задачи; **Функциональная полнота** (*functional completeness*) — определяет, насколько полно ПО способно решать нужный набор задач; **Точность** (*functional correctness*) — способность выдавать результаты с нужной точностью;

- **Производительность** (*performance efficiency*); **Временная эффективность** (*time behavior*) — способность ПО выдавать ожидаемые результаты, а также обеспечивать передачу необходимого объема данных за отведенное время; **Эффективность использования ресурсов** (*resource utilization*) — способность решать нужные задачи с использованием определенных объемов ресурсов (памяти различных видов, устройств ввода-вывода и пр.); **Пропускная способность каналов связи** (*capacity*) — способность решать нужные задачи при определенных ограничениях на пропускаемые через каналы связи объемы информации;

- **Совместимость** (*compatibility*). **Способность к сосуществованию** (*co-existence*) — из переносимости по ISO 9126, способность ПО сосуществовать с другими программами в общем окружении, деля с ними ресурсы; **Способность к взаимодействию** (*interoperability*) — из функциональности по ISO 9126, способность взаимодействовать с нужным набором других систем;

- **Удобство использования** (*usability*); **Удобство обучения** (*learnability*) — показатель, обратный усилиям, затрачиваемым пользователями на обучение выполнению

определенных задач с помощью ПО; Удобство работы (operability) — показатель, обратный усилиям, предпринимаемым пользователями для решения своих задач с помощью ПО; Понятность (теперь appropriateness recognizability) — показатель, обратный к усилиям, которые затрачиваются пользователями на восприятие основных понятий ПО и осознание их применимости для решения своих задач; Эстетичность (бывшая привлекательность, user interface aesthetics) — способность ПО быть привлекательным для пользователей, не вызывать эстетического отторжения; Защищенность от ошибок пользователей (user error protection) — способность игнорировать или исправлять определенные ошибки пользователей; Доступность (при различных способностях пользователей, accessibility) — способность поддерживать работу людей с ограниченными возможностями — при нарушении восприятия цветов и сильных дефектах зрения, некоторых нарушениях координации движений, и пр.;

- Надежность (reliability): Зрелость (maturity) — показатель, обратный частоте отказов ПО, обычно измеряется средним временем работы без сбоев и величиной, обратной вероятности возникновения отказа за данный период времени; Способность к восстановлению (recoverability) — способность восстанавливать определенный уровень работоспособности и целостность данных после отказа, при затрате определенного времени и ресурсов; Устойчивость к ошибкам (fault tolerance) — способность поддерживать заданный уровень работоспособности при отказах и некоторых нарушениях правил взаимодействия с окружением; Работоспособность (availability, иногда также переводится как доступность) — возможность ПО решать задачи и предоставлять пользователям информацию, несмотря на ненадежную работу сетей, отдельных серверов и т.д.;

- Защищенность (security): Конфиденциальность (confidentiality) — способность ПО защищать свои данные от доступа лиц, которые не имеют к ним допуска; Целостность (integrity) — способность ПО защищать свои данные от изменения теми лицами, которые не имеют на это прав; Строгое выполнение обязательств (неотвергаемость, non-repudiation) — способность дать убедительное подтверждение тому, что заданные операции действительно выполнялись авторизованными пользователями (а их результаты не были внесены каким-либо несанкционированным образом); Авторизуемость (операций, accountability) — возможность проследить, какие пользователи выполняли заданные операции; Аутентичность (authenticity) — способность дать подтверждение собственной идентичности (т.е., отсутствия подмены части кода или модулей по сравнению с проверенными и сертифицированными версиями) и идентичности пользователей (т.е., возможность дать определенные гарантии, что выступающий под некоторым идентификатором пользователя человек, это именно он, или имеющий право действовать от его имени);

- Удобство сопровождения (maintainability): Удобство проверки (testability) — показатель, обратный трудозатратам на проведение тестирования и других видов проверки того, что внесенные изменения привели к нужным результатам; o Анализируемость (analyzability) — удобство проведения анализа ошибок, дефектов и недостатков, а также удобство анализа необходимости изменений и их возможных последствий; Модифицируемость (modifiability, удобство внесения изменений + стабильность) — показатель, обратный трудозатратам на выполнение необходимых изменений и риску возникновения неожиданных эффектов после них; o Модульность (modularity) — возможность вносить изменения в отдельные модули с минимальным их влиянием на другие; Повторная используемость (reusability) — возможность использовать отдельные модули без модификации в рамках других систем;

- Переносимость (portability) o Адаптируемость (adaptability) — способность ПО приспособливаться к различным окружениям без проведения для этого действий, помимо заранее предусмотренных; Удобство замены (replaceability) — возможность применения данного ПО вместо других программных систем для решения тех же задач в

определенном окружении; Удобство установки (installability) — способность ПО быть установленным или развернутым в определенном окружении;

Требования к программному обеспечению

Перечисленные характеристики качества ПО представляют собой одну из систематик различных видов требований к программным системам. Такая систематика полезна, если необходимо выделить и проанализировать требования для сложной системы. Она позволяет организовать анализ, разбивая возможные требования и ограничения на разные группы и позволяя последовательно рассматривать различные аспекты системы. Сами требования как раз и говорят о том, какие конкретные свойства и характеристики должна иметь система, чтобы с ее помощью можно было успешно решать заданный набор задач. Требования определяют, какое поведение системы является правильным и желаемым, а какое должно рассматриваться как некорректное. Поэтому они играют первостепенную роль при тестировании — именно они и проверяются с его помощью, а ошибки связаны именно с их нарушениями. Сами требования определяются при анализе задач, стоящих перед рассматриваемой программной системой и ее окружения. Они могут извлекаться из различных источников.

При аккуратном следовании стандартам на процессы разработки ПО требования обычно фиксируются в документе, создаваемом при решении о разработке системы и называемом техническим заданием (а по-английски — requirements specification, спецификация требований). Иногда этот документ создается представителями заказчика, но часто его приходится писать самим разработчикам на основе информации, полученной из других источников.

- Важным источником требований являются стандарты, регламентирующие характеристики, функции и состав систем, работающих в определенной предметной области. Такие стандарты создаются на основе опыта, накопленного в большом количестве проектов, в ходе которых такие системы создавались или развивались, и поэтому содержат ценную информацию о желательных характеристиках таких систем. Часто работа системы связана с выполнением каких-то действий, регулируемых существующим законодательством и нормами, действующими в данной области. Нормы и законы тоже являются разновидностью действующих стандартов, хотя обычно они формулируются менее четко и однозначно, чем технические стандарты и правила.

- Разработчики сами могут осознать, что что-то должно быть сделано определенным образом — так обычно возникают внутренние ограничения задач, не соблюдая которые, невозможно решить их правильно. Чаще всего ограничения такого рода можно усмотреть при детальном анализе решаемой задачи.
- Иногда ряд требований к новой системе можно сформулировать на основе анализа уже существующих систем для решения схожих задач.

- Большая часть требований формулируется на основе явно высказываемых пожеланий пользователей системы, их руководителей, заказчиков ее разработки и других заинтересованных лиц.

- Наконец, наиболее нечетким, но, тем не менее, достаточно важным источником требований являются невысказанные явно потребности и нужды пользователей создаваемой системы, которые, несмотря на это, все же часто поддаются анализу. При извлечении требований из перечисленных выше источников их четкость и согласованность возрастают при движении от потребностей и пожеланий к стандартам и техническому заданию.

Качество ПО - это относительное понятие, которое имеет смысл только при учете реальных условий его применения, поэтому требования, предъявляемые к качеству, ставятся в соответствии с условиями и конкретной областью их применения. Оно характеризуется тремя аспектами: качество программного продукта, качество процессов ЖЦ и качество сопровождения или внедрения (рис. 1.1).



Рис. 1.1. Основные аспекты качества ПО

Аспект, связанный с процессами ЖЦ, определяет степень формализации, достоверности самих процессов ЖЦ разработки ПО, а также верификацию и валидацию промежуточных результатов на этих процессах. Поиск и устранение ошибок в готовом ПО проводится методами тестирования, которые снижают количество ошибок и повышают качество этого продукта.

Качество продукта достигается процедурами контроля промежуточных продуктов на процессах ЖЦ, проверкой их на достижение необходимого качества, а также методами сопровождения продукта. Эффект от внедрения ПС в значительной степени зависит от знаний обслуживающего персонала функций продукта и правил их выполнения. Модель качества ПО имеет следующие четыре уровня представления.

Первый уровень соответствует определению характеристик (показателей) качества ПО, каждая из которых отражает отдельную точку зрения пользователя на качество. К примеру, согласно стандарту ISO 9126 в модель качества входит шесть характеристик или шесть показателей качества:

функциональность (functionality);

надежность (realibility);

удобство (usability);

эффективность (efficiency);

сопровождаемость (maitainnability);

переносимость (portability).

Второму уровню соответствуют атрибуты для каждой характеристики качества, которые детализируют разные аспекты конкретной характеристики. Набор атрибутов характеристик качества используется при оценке качества.

Третий уровень предназначен для измерения качества с помощью метрик, каждая из них согласно стандарту определяется как комбинация метода измерения атрибута и шкалы измерения значений атрибутов. Для оценки атрибутов качества на этапах ЖЦ (при просмотре документации, программ и результатов тестирования программ) используются метрики с заданным оценочным весом для нивелирования результатов метрического анализа совокупных атрибутов конкретного показателя и качества в

целом. Атрибут качества определяется с помощью одной или нескольких методик оценки на этапах ЖЦ и на завершающем этапе разработки ПО.

Четвертый уровень - это оценочный элемент метрики (вес), который используется для оценки количественного или качественного значения отдельного атрибута показателя ПО. В зависимости от назначения, особенностей и условий сопровождения ПО выбираются наиболее важные характеристики качества и их атрибуты (рис. 1.2).

Выбранные атрибуты и их приоритеты отражаются в требованиях на разработку систем либо используется соответствующие приоритеты эталона класса ПО, к которому это ПО относится.

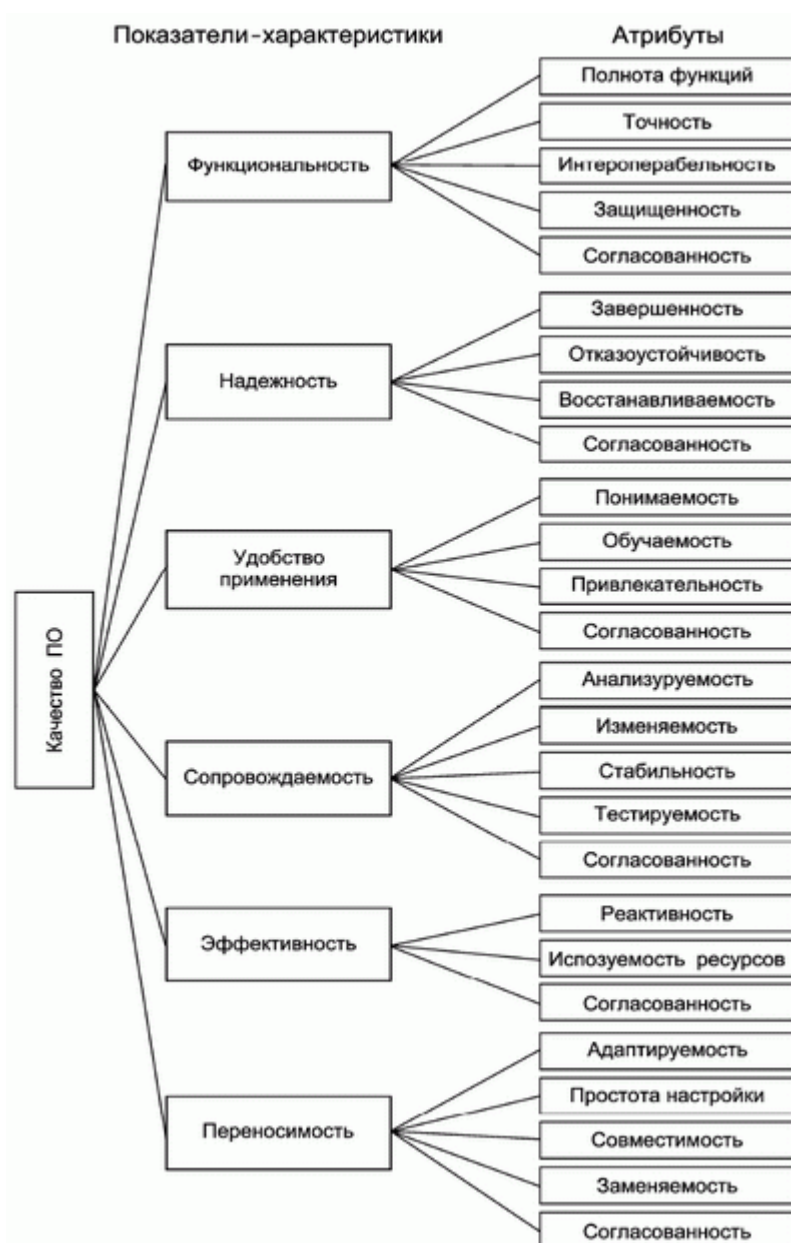


Рис. 1.2 Характеристики качества и их атрибуты.

Источники: <http://docs.cntd.ru/document/1200121069>

[https://edisciplinas.usp.br/pluginfile.php/294901/mod\\_resource/content/1/ISO%2025010%20-%20Quality%20Model.pdf](https://edisciplinas.usp.br/pluginfile.php/294901/mod_resource/content/1/ISO%2025010%20-%20Quality%20Model.pdf)

<http://ingraf.su/wp-content/uploads/GOST-R-ISO-MEK-25010-2015.pdf>

<http://prog.bobrodobro.ru/13148>