

Программное обеспечение информационных технологий

Программа – это упорядоченная последовательность команд, предназначенная для решения разных задач с помощью компьютерной техники и технологии; точная и подробная последовательность инструкций на понятном компьютеру языке с указанием правил обработки информации.

Совокупность программ, используемых при работе на компьютере, составляет его **программное обеспечение** (*software*).

Наряду с термином «*software*» для обозначения основных компонент программно-аппаратных компьютерных средств используют также термины:

Hardware – технические устройства компьютера (“железо”) или аппаратные средства, созданные, в основном, с использованием электронных и электромеханических элементов и устройств.

Brainware – знания и умения, необходимые пользователям для грамотной работы на компьютере (компьютерная культура и грамотность).

ПО ИТ и ИС можно условно разделить на три основные категории:

➤ **Системные программы** - осуществляют управление устройствами компьютера, вычислительными процессами, занимаются поиском и диагностикой неисправностей. ПО систем управления можно разделить на:

- *Операционные системы* – являются посредником между пользователем и ПК, с их помощью обеспечивается работа системных и пользовательских программ.
- *Утилиты* – программы, которые обеспечивают протекание разных вспомогательных процессов.
- *Драйверы* – программы, позволяющие ОС распознавать подключаемые внешние устройства.

➤ **Инструментальные системы** – системы, предоставляющие разработчикам приложений инструменты для работы (языки программирования, среды разработки).

➤ **Прикладные программы** - пользовательское ПО, не относящееся к двум предыдущим группам, предназначенное для решения конкретных задач пользователя (например, набор текста, воспроизведение звука или видео, создание графических файлов и т.д.).

Все прикладные программы по характеру использования и категориям пользователей можно разделить на два класса:

- **утилитарные программы** («программы для себя») предназначены для удовлетворения нужд их разработчиков. Чаще всего утилитарные программы выполняют роль сервиса в технологии обработки данных либо являются программами решения функциональных задач, не предназначенных для широкого распространения;
- **программные продукты** («изделия») предназначены для удовлетворения потребностей пользователей, широкого распространения и продажи.

Специфические виды прикладных программ:

- ***freeware*** – бесплатные программы, свободно распространяемые, поддерживаются самим пользователем, который правомочен вносить в них необходимые изменения;
- ***shareware*** – некоммерческие (условно-бесплатные) программы, которые могут использоваться бесплатно с некоторыми ограничениями (наличие рекламы, ограниченный набор функций, ограниченное время использования в течение суток и т.п.);
- ***OEM-программы*** (Original Equipment Manufacturer) - встроенные программы, поставляемые вместе с вычислительной техникой.

Программный продукт – комплекс взаимосвязанных программ для решения определенной проблемы (задачи) массового спроса, подготовленный к реализации как любой вид промышленной продукции.

Чтобы созданный программный комплекс мог быть назван программным продуктом, он должен быть соответствующим образом подготовлен к эксплуатации, иметь необходимую техническую документацию, предоставлять сервис и гарантию надежной работы программы, иметь товарный знак изготовителя, а также желательно наличие кода государственной регистрации.

Способы создания программных продуктов:

– *индивидуальная разработка под заказ*, при которой фирма-разработчик создает оригинальный программный продукт, учитывающий специфику задач конкретного заказчика;

– *разработка для массового распространения среди пользователей*, при которой фирма-разработчик, с одной стороны, должна обеспечить универсальность выполняемых функций, с другой стороны, гибкость и настраиваемость программного продукта на условия конкретного применения.

Как правило, программные продукты требуют *сопровождения*, которое осуществляется специализированными фирмами – распространителями программ (дистрибьюторами), реже – фирмами-разработчиками.

технические характеристики

объём файлов программ;

требования к операционной системе

требования к аппаратной части ПК:

- объём дисковой памяти;
- размер оперативной памяти;
- тип процессора;

необходимость установки дополнительного ПО

характеристики алгоритма

алгоритмическая сложность

состав и глубина проработки реализованных функций

полнота и системность функций

быстродействие

характеристики с точки зрения пользователя

набор доступных функций

простота и удобство пользовательского интерфейса

удобство и функциональность средств сохранения результатов

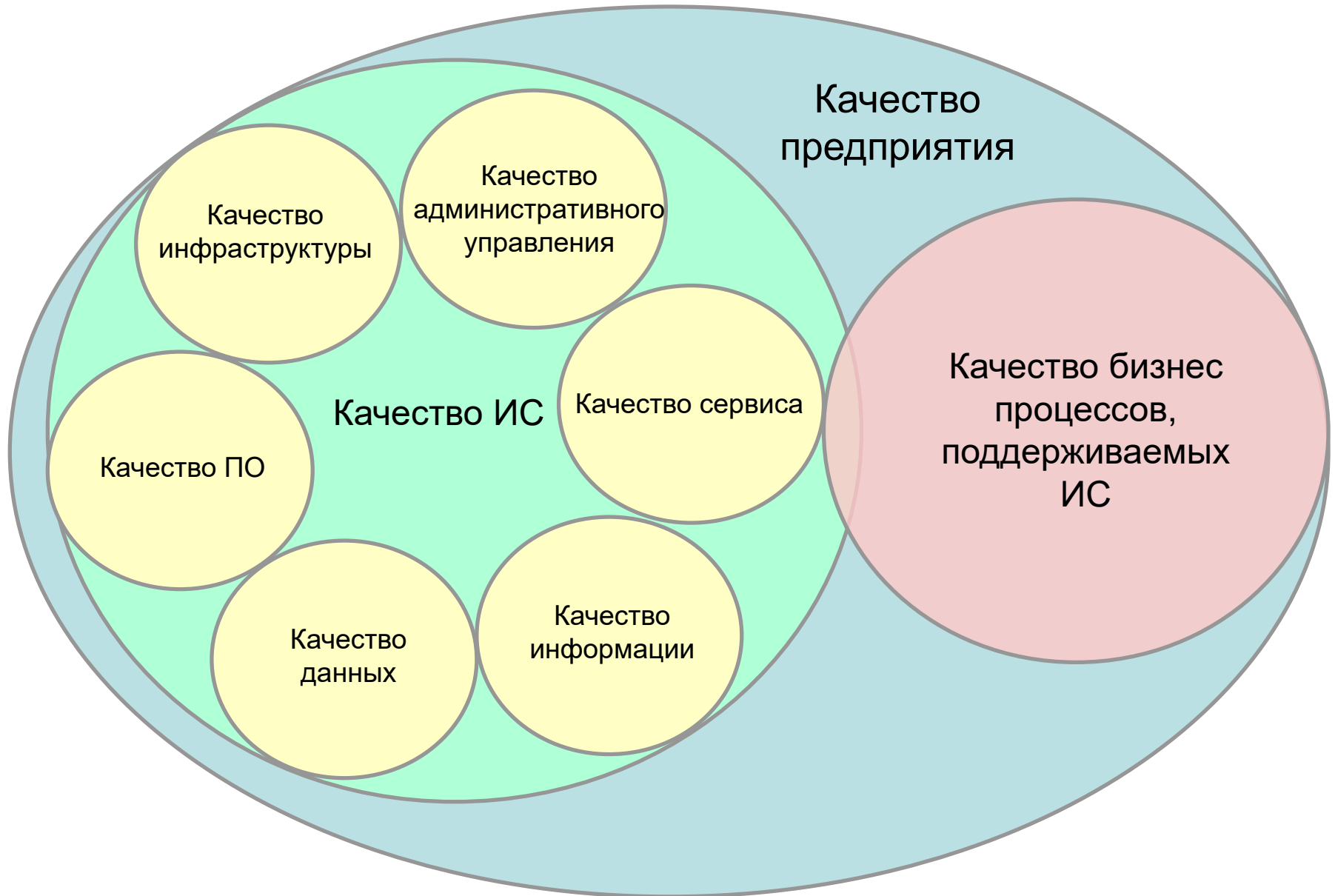
Качество программного обеспечения (Software Quality) - это степень, в которой программное обеспечение обладает требуемой комбинацией свойств.
[1061-1998 IEEE Standard for Software Quality Metrics Methodology]

Качество программного обеспечения (Software Quality) - это совокупность характеристик программного обеспечения, относящихся к его способности удовлетворять установленные и предполагаемые потребности.
[ISO 8402:1994 Quality management and quality assurance]

Обеспечение качества (Quality Assurance - QA) – это совокупность мероприятий, охватывающих все технологические этапы разработки, выпуска и эксплуатации программного обеспечения информационных систем, предпринимаемых на разных стадиях *жизненного цикла ПО*, для обеспечения требуемого уровня качества выпускаемого продукта.

Контроль качества (Quality Control - QC) - это совокупность действий, проводимых над продуктом в процессе разработки, для получения информации о его актуальном состоянии в следующих разрезах:

- "готовность продукта к выпуску",
- "соответствие зафиксированным требованиям",
- "соответствие заявленному уровню качества продукта".



Составляющие качества информационной системы:

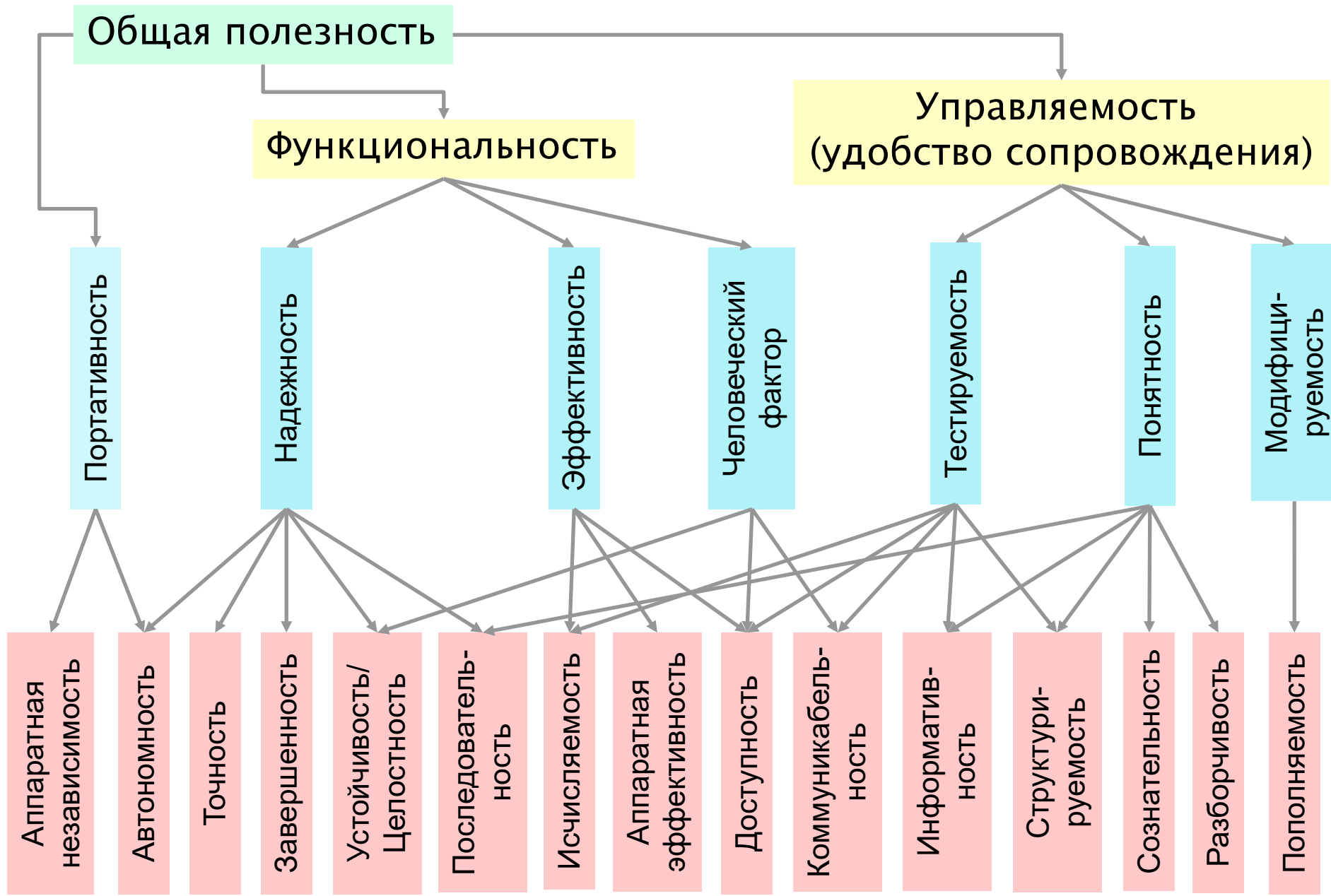
8

1. **Качество инфраструктуры** (infrastructure quality): качество аппаратного и поддерживающего программного обеспечения (например, качество операционных систем, компьютерных сетей и т.п.).
2. **Качество программного обеспечения** (software quality): качество программного обеспечения информационной системы.
3. **Качество данных** (data quality): качество данных, используемых информационной системой на входе.
4. **Качество информации** (information quality): качество информации, продуцируемой информационной системой.
5. **Качество административного управления** (administrative quality) – качество менеджмента, включая качество бюджетирования, планирования и календарного контроля.
6. **Качество сервиса** (service quality) – качество обучения, системной поддержки и т.п.

Анализ всех составляющих качества должен проводиться с учетом сфер ответственности заинтересованных сторон, как **внутренних участников исполняемого процесса** (in-process stakeholder), так и **пользователей процесса** (end-of-process stakeholders).

Управление качеством будет успешным, если под контролем находятся все измерения качества.

Дерево характеристик качества (ISO 9126)



Портативность (Portability) – характеризует ПО с точки зрения легкости его переноса из одного окружения (software/hardware) в другое (при условии смены управляющей аппаратной или программной платформы).

Функциональность (Functionality) - определяется способностью ПО решать задачи, которые соответствуют зафиксированным и предполагаемым потребностям пользователя, при заданных условиях использования ПО. Данная характеристика отвечает за то, что ПО работает исправно и точно, функционально совместимо, соответствует стандартам отрасли и защищено от несанкционированного доступа.

Удобство сопровождения (Maintainability) – легкость, с которой ПО может анализироваться, тестироваться, изменяться для исправления дефектов, для реализации новых требований, для облегчения дальнейшего обслуживания и адаптироваться к имеющемуся окружению.

Надежность (Reliability) – способность ПО выполнять требуемые задачи в обозначенных условиях на протяжении заданного промежутка времени или указанное количество операций. Атрибуты данной характеристики – это завершенность и целостность всей системы, способность самостоятельно и корректно восстанавливаться после сбоев в работе, отказоустойчивость.

Эффективность (Efficiency) – способность ПО обеспечивать требуемый уровень производительности в соответствии с выделенными ресурсами, временем и другими обозначенными условиями.

Учёт человеческого фактора (Human Engineering) - обеспечение дружественного интерфейса для работы конечного пользователя (наличие подсказки или обучающей системы, документации для освоения и использования заложенных функциональных возможностей, анализа и диагностики возникших ошибок и др.

Тестируемость (Testability) - совокупность свойств программного средства, характеризующая усилия, необходимые для его проверки после проведения какого-либо видоизменения.

Понятность (Understandability) - свойство ПО, позволяющее оценивающему лицу понять назначение программных средств.

Модифицируемость (Modifiability) – свойство ПО, характеризующее возможность внесения изменений, например расширение функций обработки, переход на другую техническую базу обработки и т.п.

Серия стандартов **ISO 9000** является наиболее широко известным и используемым стандартом для организации процессов контроля качества, однако в управлении качеством разработки программного обеспечения непосредственно использовать данный стандарт затруднительно, поскольку изначально он ориентирован на разработку промышленных изделий.

Специально для обеспечения процессов разработки программных систем организацией ISO, разработано руководство **ISO 9000-3**, которое формулирует требования модели качества **ISO 9001** к организации процесса разработки программного обеспечения. Недостатком стандарта ISO 9000 является трудность измерения уровня качества процесса разработки программного обеспечения в соответствии с предложенной моделью качества.

Среди разработчиков ПО большим рейтингом пользуется альтернативная модель качества **CMM**, разработанная в 1986 году в институте инженерии программного обеспечения (Software Engineering Institute) при спонсорстве министерства обороны США.

В определенном отношении модели качества **CMM** и **ISO** являются взаимозаменяемыми, однако, по сути, они не противоречат друг другу, поскольку основаны на одной парадигме качества – **TQM – Total Quality Management**.

Capability Maturity Model (СММ) — модель зрелости возможностей (модель 13 полноты потенциала) создания ПО. В соответствии с СММ различают 5 уровней развития предприятия:

- 1. Начальный.** Организация способна разрабатывать ПО, однако не имеет явно осознанного процесса. Качество продукта целиком определяется индивидуальными способностями разработчиков. Успех одного проекта не гарантирует успех другого. При завершении проекта не фиксируются данные о трудозатратах, расписании и качестве.
- 2. Повторяемый.** Процесс в некоторой степени отслеживается, делаются записи о трудозатратах и планах. Функциональность каждого проекта описана в письменной форме.
- 3. Установленный.** Имеется определённый, документированный и установленный процесс работы, не зависящий от отдельных личностей (вводятся согласованные профессиональные стандарты, а разработчики их выполняют). Такие организации в состоянии достаточно надёжно предсказывать затраты на проекты, аналогичные выполненным ранее.
- 4. Управляемый.** Могут точно предсказать сроки и стоимость работ. Есть база данных накопленных измерений. Но нет измерений при появлении новых технологий и парадигм.
- 5. Оптимизированный.** Есть постоянно действующая процедура поиска и освоения новых и улучшенных методов и инструментов.

Метрики менеджмента:

- Цена (Cost) – расходы на приобретение/разработку
- Время разработки (Time-to-market)
- Среда разработки (Software Engineering Environment)
- Использование системных ресурсов (System Resource Utilization)

Метрики требований (дают возможность контролировать спецификации, изменение требований, а также степень их удовлетворения):

- Соответствие требованиям (requirement conformance)
- Стабильность требований (requirement stability)

Метрики качества:

- Адаптируемость (adaptability)
- Сложность интерфейсов и интеграции (complexity of interfaces and integration)
- Тестовое покрытие (test coverage)
- Надежность (reliability)
- Профили ошибок (fault profiles)
- Степень удовлетворения потребностей заказчика (customer satisfaction)

Могут использоваться на этапах планирования и контроля проектов и других задач управления или использоваться в качестве параметров управления штатной ERP системы.

Метрика «cost» измеряет общую цену, включая цену анализа рынка, приобретения, интеграции и улучшения качества.

Метрика «time-to-market» - мера времени от формирования заказа на программу до поставки. При итерационной разработке данная метрика модифицируется для измерения времени, требуемого для поставки заданного объема приращения функциональности, то есть скорости поставки.

Метрика «System resource utilization» - определяет процент целевых компьютерных ресурсов, используемых системой.

«Software engineering environment» - мера способности производителя разрабатывать программное обеспечение высокого качества. Данная метрика может быть выражена в терминах модели «Software Acquisition Capability Maturity Model (SA – CMM).

«*Adaptability*» - мера гибкости системы, оценивает способность системы адаптироваться к изменениям требований либо перепроектированием системы, либо интеграцией приложений.

«*Complexity of interfaces and integration*» - метрика, измеряющая степень сложности интерфейса или дополнительного программирования требуемого для интеграции компоненты в систему, которые требуются для тестирования, отладки и сопровождения, компенсирующего потерю качества.

Метрики «test coverage» указывают степень полноты различных типов тестирования.

«*Reliability*»- метрика, оценивающая вероятность работы системы без отказов. Данная метрика может быть получена в рамках традиционного подхода.

«*Fault profiles*» - метрика, измеряющая кумулятивное число обнаруженных ошибок.

«*Customer satisfaction*» - метрика, оценивающая степень соответствия программного обеспечения ожиданиям и требованиям заказчика. Данная метрика может быть оценена перед поставкой на этапе опытной эксплуатации на основе прогнозирующих параметров.