

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Кафедра программного обеспечения информационных технологий

В.В. БАХТИЗИН, Л.А. ГЛУХОВА

**СТАНДАРТИЗАЦИЯ И СЕРТИФИКАЦИЯ  
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

*Допущено Министерством образования Республики Беларусь  
в качестве учебного пособия  
для студентов специальности  
«Программное обеспечение информационных технологий»  
учреждений, обеспечивающих получение высшего образования*

Минск БГУИР 2006

УДК 004.4 (075.8)  
ББК 32.973–018.2 я73  
Б 30

**Рецензенты:**

кафедра дискретной математики и алгоритмики  
Белорусского государственного университета,  
заведующий кафедрой, доктор физ.-мат. наук, профессор В.М. Котов,  
заведующий кафедрой программного обеспечения сетей электросвязи  
Высшего государственного колледжа связи,  
доктор техн. наук, профессор А.А. Прихожий

**Бахтизин В.В.**

Б 30      Стандартизация и сертификация программного обеспечения: Учеб. пособие / В.В. Бахтизин, Л.А. Глухова. – Мн.: БГУИР, 2006. – 200 с.: ил.  
ISBN 985-444-851-7

Учебное пособие посвящено изучению вопросов стандартизации и сертификации программного обеспечения (ПО). Рассмотрены стандарты в области жизненного цикла ПО: ISO/IEC 12207:1995, ISO/IEC 12207:1995/Amd.1:2002, ISO/IEC 12207:1995/Amd.2:2004, ISO/IEC TR 15271:1998, ISO/IEC TR 16326:1999, ISO/IEC 15910:1999 и их аутентичные аналоги – стандарты СТБ ИСО/МЭК и ГОСТ Р ИСО/МЭК. Рассмотрены стандарты в области оценки качества ПО: ГОСТ 28806–90, ГОСТ 28195–99, СТБ ИСО/МЭК 9126–2003, действующие на территории Республики Беларусь, и международные стандарты ISO/IEC 9126–1–4:2001–2004, ISO/IEC 14598–1–6:1999–2001. Рассмотрены вопросы сертификации ПО. Даны сведения о Национальной системе оценки соответствия Республики Беларусь.

Учебное пособие предназначено для студентов специальности «Программное обеспечение информационных технологий» высших учебных заведений, изучающих дисциплину «Стандартизация и сертификация программного обеспечения» и родственные дисциплины.

**УДК 004.4 (075.8)**  
**ББК 32.973–018.2 я73**

ISBN 985-444-851-7

© Бахтизин В.В., Глухова Л.А., 2006  
© БГУИР, 2006

# Содержание

Введение.....	7
<b>Часть I. Стандартизация жизненного цикла программного обеспечения .....</b>	<b>11</b>
<b>1. Жизненный цикл программных средств и систем. Основные понятия и определения .....</b>	<b>12</b>
<b>2. Стандарт ISO/IEC 12207 – базовый стандарт в области жизненного цикла программных средств и систем .....</b>	<b>15</b>
2.1. Общие сведения .....	15
2.2. Основные процессы жизненного цикла .....	19
2.2.1. Процесс заказа (The Acquisition Process).....	20
2.2.2. Процесс поставки (The Supply Process) .....	22
2.2.3. Процесс разработки (The Development Process).....	26
2.2.4. Процесс эксплуатации (The Operation Process).....	37
2.2.5. Процесс сопровождения (The Maintenance Process) .....	38
2.3. Вспомогательные процессы жизненного цикла.....	40
2.3.1. Процесс документирования (The Documentation Process).....	41
2.3.2. Процесс управления конфигурацией (The Configuration Management Process) .....	42
2.3.3. Процесс обеспечения качества (The Quality Assurance Process) .....	44
2.3.4. Процесс верификации (The Verification Process) .....	46
2.3.5. Процесс аттестации (The Validation Process) .....	50
2.3.6. Процесс совместного анализа (The Joint Review Process) .....	52
2.3.7. Процесс аудита (The Audit Process) .....	53
2.3.8. Процесс решения проблем (The Problem Resolution Process).....	55
2.4. Организационные процессы жизненного цикла .....	56
2.4.1. Процесс управления (The Management Process).....	56
2.4.2. Процесс создания инфраструктуры (The Infrastructure Process) .....	59
2.4.3. Процесс усовершенствования (The Improvement Process).....	60
2.4.4. Процесс обучения (The Training Process) .....	61
2.5. Адаптация требований стандарта ИСО/МЭК 12207 к условиям проекта .....	62
2.6. Инструментальные средства автоматизации жизненного цикла программных средств и систем.....	66
2.6.1. Telelogic DOORS.....	67
2.6.2. Telelogic TAU.....	68
2.6.3. Telelogic SYNERGY .....	70
2.6.4. Telelogic DocExpress .....	71
2.6.5. Telelogic TAU Logiscope.....	71

<b>3. Дополнения ISO/IEC 12207:1995 / Amd.1 и ISO/IEC 12207:1995 / Amd.2</b>	<b>73</b>
3.1. Общие сведения	73
3.2. Новые процессы жизненного цикла программных средств	76
3.2.1. Процесс практичности (The Usability Process)	76
3.2.2. Процесс человеческого ресурса (The Human Resource Process)	76
3.2.3. Процесс управления средствами (The Asset Management Process)	78
3.2.4. Процесс управления повторным использованием программ (The Reuse Program Management Process)	78
3.2.5. Процесс проектирования предметной области (The Domain Engineering Process)	79
3.3. Расширенные процессы жизненного цикла программных средств	80
3.3.1. Процесс оценки продукта (The Product Evaluation Process)	80
3.3.2. Процесс управления решением проблем (The Problem Resolution Management Process)	80
3.3.3. Процесс управления заявками на изменения (The Change Request Management Process)	81
3.4. Процессы жизненного цикла с измененной структурой	82
3.4.1. Процесс заказа (The Acquisition Process)	82
3.4.2. Процесс поставки (The Supply Process)	82
3.4.3. Процесс разработки (The Development Process)	83
3.4.4. Процесс эксплуатации (The Operation Process)	85
3.4.5. Процесс управления (The Management Process)	86
3.4.6. Расширенное описание процесса заказа	88
<b>4. Стандарт ISO/IEC TR 15271</b>	<b>90</b>
4.1. Общие сведения	90
4.2. Основные концепции стандарта ИСО/МЭК 12207	90
4.3. Внедрение ИСО/МЭК 12207	93
4.4. Применение ИСО/МЭК 12207 в проектах и организациях	94
4.5. Прикладное применение модели жизненного цикла системы	95
4.6. Процессы качества и требования к оценке	98
4.7. Модели жизненного цикла	99
4.7.1. Каскадная модель	99
4.7.2. Инкрементная модель	101
4.7.3. Эволюционная модель	103
4.8. Примеры адаптации ИСО/МЭК 12207	104
4.8.1. Пример макетирования небольшой системы	105
4.8.2. Пример ускоренной разработки приложения	106
<b>5. Стандарт ISO/IEC TR 16326</b>	<b>109</b>
5.1. Общие сведения	109
5.2. Основные определения	110
5.3. Руководство по процессу управления	112

<b>6. Стандарт ISO/IEC 15910</b>	119
6.1. Общие сведения	119
6.2. Основные определения	120
6.3. Требования к процессу документирования	120
6.4. Содержание плана документирования	125
6.5. Требования к содержанию спецификации стиля документации	126
<b>7. Стандарт ISO/IEC 15288</b>	128
7.1. Общие сведения	128
7.2. Процессы жизненного цикла систем	129
Вопросы для самопроверки к части I	134
<b>Часть II. Стандартизация оценки качества программного обеспечения</b>	136
<b>8. Качество программных средств. Основные понятия и определения</b>	137
<b>9. Стандартизация качества программного обеспечения в Республике Беларусь</b>	140
9.1. Общие сведения о стандартах в области оценки качества, действующих на территории Республики Беларусь	140
9.2. Классификация методов определения показателей качества программного средства	141
9.3. Иерархическая модель оценки качества программного средства	142
9.4. Метод оценки качества программных средств по ГОСТ 28195–99	143
9.5. Метод оценки качества программных средств по СТБ ИСО/МЭК 9126–2003	153
<b>10. Стандартизация качества программных средств за рубежом</b>	156
10.1. Стандарты серии ISO/IEC в области оценки качества программных средств	156
10.2. Связь качества программного средства с его жизненным циклом	159
10.3. Модель внешнего и внутреннего качества ПС	160
10.3.1. Функциональность	163
10.3.2. Надежность	163
10.3.3. Практичность	164
10.3.4. Эффективность	164
10.3.5. Сопровождаемость	165
10.3.6. Мобильность	165
10.4. Модель качества в использовании	166
10.5. Метрики качества программных средств	167
10.5.1. Свойства и критерии обоснованности метрик	167
10.5.2. Внутренние метрики качества программных средств	170

10.5.3. Внешние метрики качества программных средств .....	175
10.5.4. Метрики качества программных средств в использовании .....	179
10.6. Метод оценки качества программных средств по ISO/IEC 14598–1:1999 .....	180
Вопросы для самопроверки к части II .....	185
<b>Часть III. Сертификация программного обеспечения .....</b>	<b>186</b>
<b>11. Сертификация программных средств .....</b>	<b>187</b>
11.1. Основные понятия и определения в области технического нормирования, стандартизации и оценки соответствия .....	187
11.2. Общие сведения об оценке соответствия в Республике Беларусь .....	189
11.3. Организация сертификации программных средств .....	195
Вопросы для самопроверки к части III .....	197
<b>Литература .....</b>	<b>198</b>

# ВВЕДЕНИЕ

Широкое внедрение информационных технологий во все сферы деятельности человека приводит к разработке огромного количества программных средств (ПС) различного функционального назначения. Это вызывает необходимость в создании международных и национальных стандартов в области программных средств, систем и связанных с ними процессов. Применение стандартов позволяет унифицировать процессы разработки, эксплуатации и сопровождения ПС, что способствует повышению качества процессов и самих ПС и повышению конкурентоспособности ПС на внутреннем и внешнем рынках.

Основной международной организацией, занимающейся деятельностью по стандартизации, является *Международная организация по стандартизации ИСО* (International Standards Organization, ISO). Стандарты, принимаемые данной организацией, получают аббревиатуру ISO и имеют ранг международных.

Области электротехники, электроники, радиосвязи, приборостроения не входят в компетенцию ISO. Стандартизацией в данных областях занимается *Международная электротехническая комиссия МЭК* (International Electrotechnical Commission, IEC). В области стандартизации информационных технологий ISO и IEC объединили свою деятельность, создав *Объединенный технический комитет 1* (Joint Technical Committee 1, JTC1).

В Республике Беларусь республиканским органом государственного управления по вопросам технического нормирования, стандартизации, метрологии и оценки соответствия является Комитет по стандартизации, метрологии и сертификации при Совете Министров Республики Беларусь (*Госстандарт*).

В систему Госстандарта входит *Белорусский государственный институт стандартизации и сертификации (БелГИСС)*, занимающийся вопросами технического нормирования, стандартизации, подтверждения соответствия, управления качеством, нормативно-технического и информационного обеспечения.

За рубежом и в Беларуси приняты следующие **обозначения стандартов**:

- вначале записывается категория стандарта; например, СТБ – стандарт Беларуси, ГОСТ Р – государственный стандарт России, ГОСТ – межгосударственный стандарт для ряда стран СНГ, до распада СССР аббревиатура ГОСТ обозначала государственный стандарт СССР, ISO/IEC (ИСО/МЭК) – международный стандарт организаций ISO и IEC;
- если стандарт разработан методом прямого применения (например, является аутентичным переводом международного стандарта), то за категорией стандарта следует обозначение категории базового стандарта (в русском именовании); например, СТБ ИСО/МЭК и ГОСТ Р ИСО/МЭК – это аутентичные переводы международного стандарта ISO/IEC, СТБ ГОСТ Р – стандарт Республики Беларусь, разработанный методом прямого применения стандарта России;
- затем следует номер стандарта; при этом, если стандарт разработан методом прямого применения, то его номер совпадает с номером базового стандарта; например, базовый стандарт ISO/IEC 9126, аутентичные стандарты Бела-

руси СТБ ИСО/МЭК 9126 и России ГОСТ Р ИСО/МЭК 9126;

- если стандарт состоит из нескольких частей, то после номера стандарта записывается номер его части; например, ISO/IEC 14598–1;
- после номера стандарта (или его части) записывается год его утверждения; например, ISO/IEC 9126:1991, СТБ ИСО/МЭК 9126–2003;
- если стандарт еще находится в стадии разработки, но имеется необходимость в опубликовании его материалов, то в обозначении стандартов ISO/IEC, работа над которыми начата до 1999 г., после категории записывается аббревиатура TR (Technical Report – Технический отчет, ТО); например, ISO/IEC TR 15271:1998, ГОСТ Р ИСО/МЭК ТО 15271–2002.

Существующие стандарты можно разделить на *два основных типа* [40]: *стандарты на продукты*, определяющие их характеристики и требования к ним; *стандарты на процессы*, определяющие конкретные методы создания продуктов. В пособии рассматриваются стандарты обоих типов.

Учебное пособие состоит из трех частей.

**Часть I** пособия посвящена изучению современных стандартов в области жизненного цикла (ЖЦ) ПС и систем, действующих за рубежом и в Беларуси. Стандартизация процессов ЖЦ ПС и систем занимает важное место в стандартизации информационных технологий и программной инженерии. Строгое соблюдение стандартов, связанных с ЖЦ ПС, обеспечивает улучшение технико-экономических показателей проектов ПС, позволяет унифицировать процесс и технологии разработки ПС. Это приводит к существенному повышению качества как процессов ЖЦ ПС, так и самих программных продуктов (ПП).

В **первом разделе** рассматриваются основные понятия и определения в области ЖЦ ПС и систем.

Во **втором разделе** детально рассмотрены и пояснены требования и рекомендации базового стандарта в области ЖЦ ПС и систем *ISO/IEC 12207:1995* и его аутентичных аналогов – стандартов Беларуси *СТБ ИСО/МЭК 12207–2003* и России *ГОСТ Р ИСО/МЭК 12207–99* [17, 12, 5]. Рассмотрены процессы, работы и задачи ЖЦ ПС и систем. Пояснены возможности адаптации требований стандарта *ИСО/МЭК 12207* к условиям конкретного проекта.

В настоящее время базовый стандарт *ISO/IEC 12207:1995* развивается в двух направлениях. *Первое направление* связано с добавлением новых процессов и модификацией структуры некоторых существующих процессов ЖЦ. *Второе направление* связано с детализацией описания существующих процессов ЖЦ, определенных в *ISO/IEC 12207:1995*. Оба направления детально отражены в следующих разделах данной части пособия.

**Третий раздел** посвящен рассмотрению изменений, внесенных в стандарт *ISO/IEC 12207:1995* Дополнениями *ISO/IEC 12207:1995/Amd.1:2002* и *ISO/IEC 12207:1995/Amd.2:2004*. Дополнения учитывают современный взгляд на стандартизацию ЖЦ ПС и систем и необходимость координации с другими действующими стандартами. Проанализированы новые процессы и подпроцессы ЖЦ ПС и систем, а также процессы, в структуру которых внесены изменения.



**Четвертый раздел** посвящен изучению международного стандарта *ISO/IEC TR 15271:1998* (аутентичный стандарт России *ГОСТ Р ИСО/МЭК ТО 15271–2002*) [28, 8]. Стандарт содержит рекомендации по практическому применению базового стандарта *ИСО/МЭК 12207* в условиях реализации конкретных проектов создания ПС. Анализируются *три базовые* стратегии разработки ПС (каскадная, инкрементная, эволюционная). Приведены примеры адаптации стандарта *ИСО/МЭК 12207* и его конкретного использования.

В **пятом разделе** рассматривается международный стандарт *ISO/IEC TR 16326:1999* (стандарт России *ГОСТ Р ИСО/МЭК ТО 16326–2002*) [43, 9]. Стандарт содержит рекомендации по эффективной реализации задач процесса управления, описанного в стандарте *ИСО/МЭК 12207*. Детализируется содержание планов управления программным проектом и методов его контроля.

В **шестом разделе** рассмотрен международный стандарт *ISO/IEC 15910:1999* (стандарт России *ГОСТ Р ИСО/МЭК 15910–2002*) [40, 7]. В стандарте описан метод применения процесса документирования из стандарта *ИСО/МЭК 12207* при создании конкретного ПС. Приведены структура процесса документирования и комплексного плана разработки документации.

В **седьмом разделе** кратко рассмотрен базовый международный стандарт *ISO/IEC 15288:2002* [29] в области ЖЦ систем, состоящих из аппаратных средств, программного обеспечения, персонала и процессов. Описан набор из двадцати пяти процессов ЖЦ, объединенных в четыре группы. Определена связь стандартов *ISO/IEC 15288:2002* и *ISO/IEC 12207:1995*.

Помимо рассмотренных в части I пособия стандартов интерес в области ЖЦ ПС и систем представляют следующие стандарты.

Международный стандарт *ISO/IEC 14764:1999* (стандарты Беларуси *СТБ ИСО/МЭК 14764–2003* и России *ГОСТ Р ИСО/МЭК 14764–2002*) [27, 13, 6]. Стандарт детализирует описание процесса сопровождения, установленного в стандарте *ИСО/МЭК 12207*, уточняет основные требования к данному процессу, условия его реализации, выходные результаты его работ.

Международный стандарт *ISO/IEC 16085:2004* [42]. Данный стандарт детализирует описание подпроцесса управления рисками, входящего в состав модифицированной структуры процесса управления, определенной в Дополнении *ISO/IEC 12207:1995 / Amd.1:2002*.

Международный стандарт *ISO/IEC 15939:2002* [41]. Данный стандарт содержит подробное описание деятельности по измерению, включенной в качестве нового подпроцесса модифицированного процесса управления, который определен в Дополнении *ISO/IEC 12207:1995 / Amd.1:2002*.

Американский национальный стандарт *IEEE 1517–1999* [15]. Требования и рекомендации данного стандарта являются основой для определенных в Дополнении *ISO/IEC 12207:1995 / Amd.1:2002* новых организационных процессов ЖЦ ПС – процессов управления средствами, управления повторным использованием программ и проектирования предметной области.

Международный стандарт *ISO/IEC 15846:1998* [39]. В данном стандарте приведены рекомендации по эффективной реализации процесса управления

конфигурацией, определенного в *ИСО/МЭК 12207*.

Международный стандарт *ISO/IEC TR 19760:2003* [44]. Данный стандарт содержит рекомендации по практическому применению стандарта *ISO/IEC 15288:2002* в условиях реализации конкретных проектов создания систем.

**Часть II** пособия посвящена изучению современных стандартов в области оценки качества ПС. Применение компьютеров становится все разнообразнее. Их корректная работа, определяемая в первую очередь программным обеспечением, часто является критичной для здоровья и безопасности человека, успеха предприятий и организаций. В этой связи первостепенное значение имеет разработка ПП высокого качества, ключевыми факторами которого являются тщательное специфицирование и оценка качества промежуточных и конечного ПП.

В **восьмом разделе** рассматриваются основные понятия и определения в области качества программных средств.

**Девятый раздел** посвящен изучению основных положений стандартов в области оценки качества ПС, действующих на территории Республики Беларусь. Рассмотрены требования и рекомендации стандартов *ГОСТ 28806–90*, *ГОСТ 28195–99*, *СТБ ИСО/МЭК 9126–2003* [3, 2, 14]. Рассмотрены модели и методы оценки качества ПС, определенные в данных стандартах.

В **десятом разделе** рассматриваются современные международные стандарты в области оценки качества ПС. Основными из них являются серии стандартов *ISO/IEC 9126–1–4:2001–2004* и *ISO/IEC 14598–1–6:1998–2001* [49 – 52, 21 – 26]. В стандартах *ISO/IEC 9126–1–4:2001–2004* регламентируется иерархическая модель качества ПС, состоящая из характеристик, подхарактеристик и метрик. В стандартах *ISO/IEC 14598–1–6:1999–2001* определен процесс оценки качества ПС и его модификации для различных целей оценки.

**Часть III** пособия посвящена вопросам сертификации ПС. Сертификация ПС предназначена для обеспечения защиты жизни и здоровья человека, защиты имущества, охраны окружающей среды при использовании ПС, для повышения конкурентоспособности ПС и создания благоприятных условий для обеспечения свободного перемещения ПС на внутреннем и внешнем рынках.

**Одиннадцатый раздел** посвящен сертификации программных средств. Рассмотрено положение дел в Республике Беларусь в области оценки соответствия продукции требованиям технических нормативных правовых актов. Описаны основные законы Республики Беларусь, посвященные техническому нормированию, стандартизации и оценке соответствия. Рассмотрены формы подтверждения соответствия (обязательная и добровольная сертификация и декларирование соответствия). Описана организация сертификации ПС.

**В учебном пособии используются следующие сокращения:**

ВТО – Всемирная торговая организация;

ЖЦ – жизненный цикл;

НСПС РБ – Национальная система подтверждения соответствия Республики Беларусь;

ПП – программный продукт;

ПС – программное средство.

# **ЧАСТЬ I**

## **СТАНДАРТИЗАЦИЯ ЖИЗНЕННОГО ЦИКЛА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

# 1. ЖИЗНЕННЫЙ ЦИКЛ ПРОГРАММНЫХ СРЕДСТВ И СИСТЕМ. ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ

Одним из базовых понятий в области информационных технологий является понятие жизненного цикла программных средств (ПС) и систем.

Под **жизненным циклом** (ЖЦ) подразумевается совокупность процессов, работ и задач, включающая в себя разработку, эксплуатацию и сопровождение ПС, начиная с анализа его концепции или потребности в заказе до прекращения его использования.

С понятием ЖЦ ПС или системы тесно связано понятие модели жизненного цикла. **Модель жизненного цикла** – это совокупность процессов, работ и задач ЖЦ, отражающая их взаимосвязь и последовательность выполнения.

В настоящее время во всем мире ведутся активные работы в направлении стандартизации ЖЦ ПС и систем. Стандартизация ЖЦ позволяет упорядочить вопросы создания, сопровождения и управления ПС. Строгое соблюдение требований стандартов обеспечивает унификацию процессов ЖЦ программных средств и их компонентов. Это ведет к повышению качества процессов ЖЦ и в конечном итоге к повышению качества самих программных средств.

С понятием ЖЦ ПС тесно связаны следующие **основные термины**, соответствующие определениям стандартов *ISO/IEC 2382-1:1993*, *ISO/IEC 2382-20:1990*, *ISO 8402:1994*, *СТБ ИСО/МЭК 12207-2003* [45, 46, 47, 12].

**Аттестация (validation)**: подтверждение экспертизой и представлением объективных доказательств того, что конкретные требования к конкретным объектам полностью реализованы. В процессе проектирования и разработки аттестация связана с экспертизой продукта в целях определения его соответствия потребностям (требованиям) пользователя.

**Аудит (audit)**: проверка, выполняемая компетентным органом (лицом) с целью обеспечения независимой оценки степени соответствия программных продуктов или процессов установленным требованиям.

**Базовая линия (baseline)**: официально принятая версия элемента конфигурации, независимая от среды, формально обозначенная и зафиксированная в конкретный момент времени жизненного цикла элемента конфигурации.

**Верификация (verification)**: подтверждение экспертизой и представлением объективных доказательств того, что конкретные требования полностью реализованы. В процессе проектирования и разработки верификация связана с экспертизой результатов данной работы для определения их соответствия установленным к данной работе требованиям.

**Квалификационное испытание (qualification testing)**: испытание (тестиро-

вание), проводимое разработчиком, при необходимости санкционированное заказчиком, для демонстрации того, что программный продукт удовлетворяет установленным требованиям и готов к использованию в заданных условиях эксплуатации.

*Квалификационное требование (qualification requirement):* набор критериев или условий, которые должны быть удовлетворены для того, чтобы квалифицировать программный продукт на соответствие установленным требованиям и готовность к использованию в заданных условиях эксплуатации.

*Контекст использования (context of use):* пользователи, задания, среда (аппаратное обеспечение, программное обеспечение и материалы), а также физические и социальные среды, в которых используется данное программное средство.

*Обеспечение качества (quality assurance):* все запланированные и систематически выполняемые в рамках системы качества работы; при необходимости объективные доказательства, обеспечивающие уверенность в том, что объект будет полностью соответствовать установленным требованиям качества. Существуют как внешние, так и внутренние цели обеспечения качества. Внутреннее обеспечение качества создает уверенность у руководства в достижении заданных требований качества внутри организации. Внешнее обеспечение качества создает уверенность у потребителя или других лиц в достижении заданных требований качества в договорных или других ситуациях.

*Программная услуга (software service):* выполнение работ, заданий или обязанностей, связанных с программным продуктом, таких, как разработка, сопровождение или эксплуатация.

*Программное обеспечение (программное средство, software):* полный набор (программное обеспечение) или часть (программное средство) программ, процедур, правил и связанной с ними документации системы обработки информации.

*Программный модуль (software unit):* отдельно компилируемая часть программного кода (программы).

*Программный продукт (software product):* набор компьютерных программ, процедур, связанных с ними документации и данных. Продукты включают промежуточные продукты и продукты, предназначенные для пользователей типа разработчиков и персонала сопровождения.

*Система (system):* комплекс, состоящий из процессов, технических и программных средств, устройств и персонала, обладающий возможностью удовлетворять установленным потребностям или целям.

*Система управления качеством (система менеджмента качества, система качества, quality management system, quality system):* часть общей системы управления, включающая организационную структуру, планирование, ответственность, методы, процедуры, процессы, ресурсы, необходимые для обеспечения качества продукции и (или) услуг. Как правило, система управления качеством является частью системы управления предприятием или организацией.

*Спецификация требований к программному продукту (software product specification)*: определение и перечень требований к программному продукту.

*Тестируемость (testability)*: степень, до которой могут быть запланированы объективность и реализуемость тестирования, проверяющего соответствие требованию.

*Тестовое покрытие (test coverage)*: степень, до которой с помощью контрольных примеров проверяют требования к системе или программному продукту.

*Техническое задание (statement of work)*: документ, используемый заказчиком в качестве средства для описания и определения задач, выполняемых при реализации договора.

*Элемент конфигурации (configuration item)*: объект внутри конфигурации, который удовлетворяет функции конечного использования и может быть однозначно определен в данной эталонной точке.

## 2. СТАНДАРТ ISO/IEC 12207 – БАЗОВЫЙ СТАНДАРТ В ОБЛАСТИ ЖИЗНЕННОГО ЦИКЛА ПРОГРАММНЫХ СРЕДСТВ И СИСТЕМ

### 2.1. Общие сведения

На уровне международных стандартов жизненный цикл сложных программных средств наиболее полно отражен в международном стандарте *ISO/IEC 12207:1995 – Информационная технология – Процессы жизненного цикла программных средств* [17].

В России в 2000 г. введен в действие *ГОСТ Р ИСО/МЭК 12207–99*, содержащий полный аутентичный текст международного стандарта [5].

В Республике Беларусь этот стандарт введен в действие в 2004 г. под обозначением *СТБ ИСО/МЭК 12207–2003* [12].

Стандарт *ИСО/МЭК 12207* имеет следующую структуру:

Раздел 1 – Область применения.

Раздел 2 – Нормативные ссылки.

Раздел 3 – Определения.

Раздел 4 – Прикладное применение настоящего стандарта.

Раздел 5 – Основные процессы жизненного цикла.

Раздел 6 – Вспомогательные процессы жизненного цикла.

Раздел 7 – Организационные процессы жизненного цикла.

Приложение А. Процесс адаптации.

Приложение В. Руководство по адаптации.

Приложение С. Руководство по процессам и организациям.

Приложение D. Библиография.

Данный стандарт определяет жизненный цикл ПС и систем в целом. При этом под *системой* подразумевается комплекс, состоящий из процессов, технических и программных средств и персонала, обладающий возможностью удовлетворять установленным потребностям или целям. Однако стандарт *ИСО/МЭК 12207* охватывает процессы ЖЦ системы только в части ее программных средств и не определяет процессы ЖЦ для ее остальных компонентов.

В соответствии с данным стандартом ЖЦ ПС и систем имеет трехуровневую иерархическую структуру (рис. 1). Основу жизненного цикла составляет набор *процессов*. Каждый процесс разделен на набор *работ*. Каждая работа разделена на набор *задач*. Общее число процессов в ЖЦ ПС равно 17, работ – 74, задач – 232.

Процессы ЖЦ ПС делятся на *три группы* (рис. 2):

- основные;
- вспомогательные;
- организационные.

Нумерация групп процессов на рис. 2 приведена в соответствии со структурой стандарта ИСО/МЭК 12207.

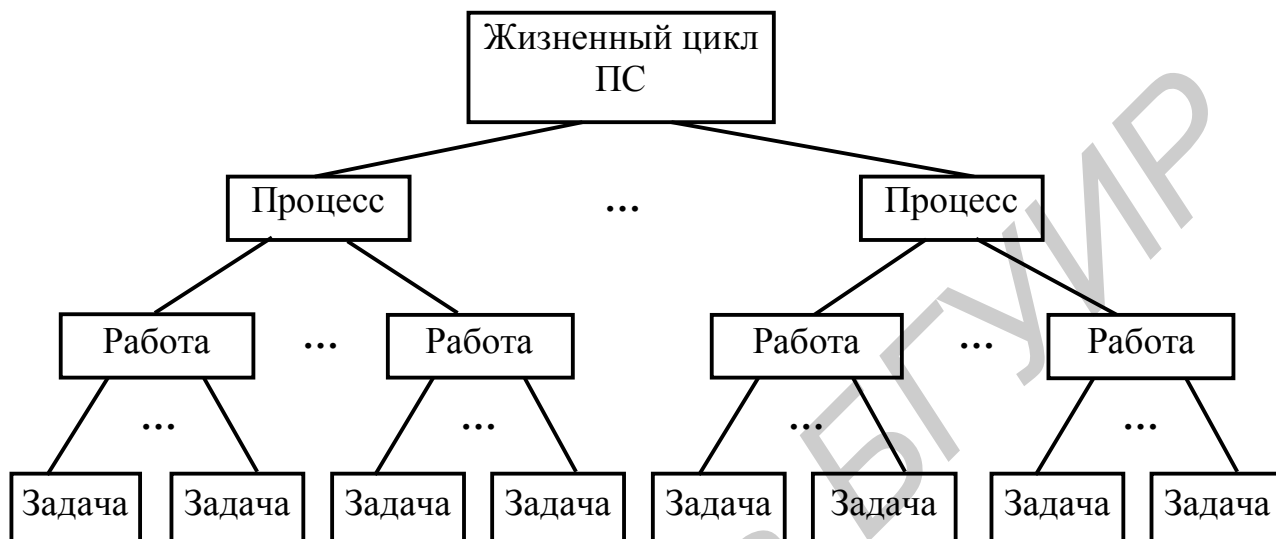


Рис. 1. Общая структура жизненного цикла программных средств

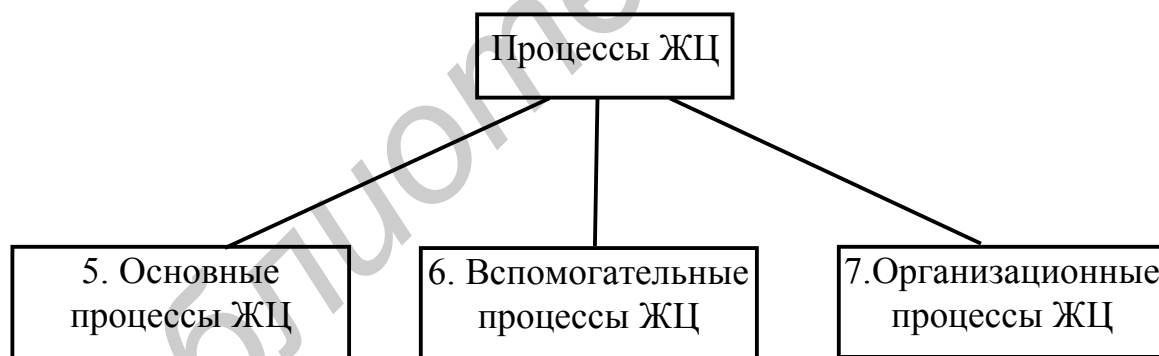


Рис. 2. Группы процессов жизненного цикла

На рис. 3 изображены процессы ЖЦ ПС с учетом их распределения по группам [12].

**Основные процессы жизненного цикла** – это процессы, которые реализуются под управлением основных сторон, участвующих в ЖЦ ПС. Основными сторонами являются заказчик, поставщик, разработчик, оператор и персонал сопровождения программных продуктов.



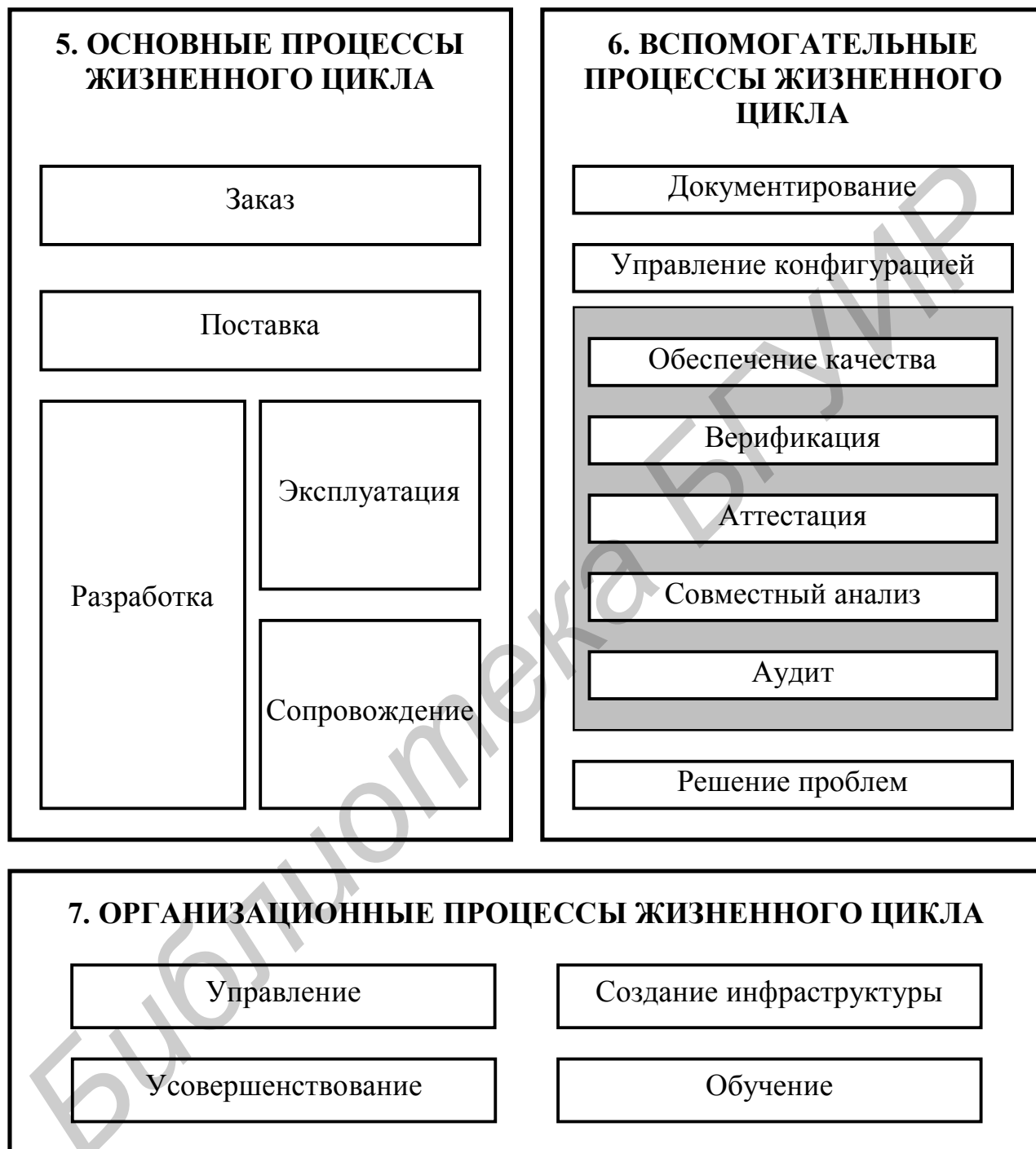


Рис. 3. Процессы жизненного цикла программных средств

**Заказчик** – это организация, которая приобретает систему, программный продукт (ПП) или программную услугу.

**Поставщик** – это организация, которая поставяет систему, ПП или программную услугу заказчику.

**Разработчик** – это организация, которая разрабатывает ПП.

**Оператор** – это организация, которая производит эксплуатационное обслуживание системы, содержащей ПП, в заданных условиях.

**Персонал сопровождения** – это организация, которая предоставляет услуги по сопровождению программного продукта.

Как видно из рис. 3, основные процессы состоят из *пяти процессов*:

- заказ;
- поставка;
- разработка;
- эксплуатация;
- сопровождение.

**Вспомогательные процессы жизненного цикла** – это процессы, являющиеся целенаправленными составными частями других процессов. Их основное назначение – обеспечить успешную реализацию и качество выполнения программного проекта. Вспомогательный процесс инициируется и используется другим процессом.

Вспомогательные процессы состоят из *восьми процессов*:

- документирование;
- управление конфигурацией;
- обеспечение качества;
- верификация;
- аттестация;
- совместный анализ;
- аудит;
- решение проблем.

Для управления качеством программных средств в ходе жизненного цикла служат процессы обеспечения качества, верификации, аттестации, совместного анализа и аудита. Данные процессы на рис. 3 выделены серым прямоугольником. При этом процессы верификации, аттестации, совместного анализа и аудита могут реализовываться различными сторонами независимо или использоваться как методы процесса обеспечения качества.

Ответственность за работы и задачи вспомогательного процесса несет организация, выполняющая данный процесс.

**Организационные процессы жизненного цикла** – это процессы, предназначенные для создания в некоторой организации и совершенствования организационных структур, охватывающих процессы ЖЦ и соответствующий персонал. Обычно организационные процессы являются типовыми.

К организационным процессам относятся *четыре процесса*:

- управление;
- создание инфраструктуры;

- усовершенствование;
- обучение.

Ответственность за работы и задачи организационного процесса несет организация, выполняющая данный процесс.

Управление процессами жизненного цикла на проектном уровне осуществляется в соответствии с процессом управления. Управление этими процессами на организационном уровне выполняется в соответствии с процессами усовершенствования и обучения. Инфраструктура процессов ЖЦ определяется в соответствии с процессом создания инфраструктуры.

Адаптация данных процессов к условиям проекта осуществляется в соответствии с процессом адаптации (см. подразд. 2.5).

Действия по управлению, созданию инфраструктуры и адаптации процессов жизненного цикла выполняют те организации, которые реализуют соответствующие процессы ЖЦ.

Следует обратить внимание на то, что взаимоотношения между процессами, определяемые в *ИСО/МЭК 12207*, всегда статические. В реальной же жизни отношения между процессами и участниками программного проекта являются динамическими. Каждый процесс и выполняющая его организация включаются в проект уникальным образом [12].

*Процесс заказа и заказчик* включаются в проект при определении системы, которая должна содержать программный продукт.

*Процесс поставки и поставщик* включаются при предоставлении программного продукта или услуги, от которых зависит система.

*Процесс разработки и разработчик* включаются в проект при анализе системы для корректного выделения и определения программного продукта, при его разработке и обеспечении подключения к системе.

*Процесс эксплуатации и оператор* включаются при эксплуатации программного продукта в системной среде в интересах пользователей.

*Процесс сопровождения и персонал сопровождения* включаются при сопровождении и поддержке программного продукта в эксплуатационной готовности и для консультаций пользователей.

*Вспомогательный или организационный процесс* включается при необходимости обеспечения уникальных специализированных функций для других процессов.

## **2.2. Основные процессы жизненного цикла**

Основные процессы жизненного цикла программных средств и систем представлены на рис. 4. На данном рисунке сохранена нумерация процессов, принятая в *ИСО/МЭК 12207*.

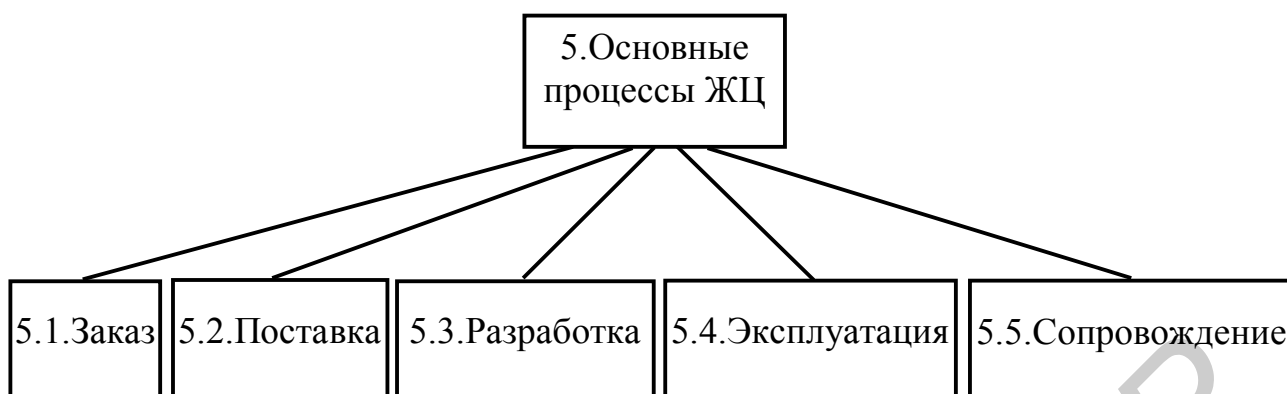


Рис. 4. Основные процессы жизненного цикла программных средств и систем

### 2.2.1. Процесс заказа (The Acquisition Process)

*Процесс заказа* определяет работы и задачи заказчика. Процесс заказа состоит из определения потребностей заказчика в системе, программном продукте или программной услуге, подготовки и выпуска заявки на подряд, выбора поставщика и управления процессом заказа до завершения приемки системы, программного продукта или программной услуги.

Процесс заказа состоит из *пяти работ* (рис. 5). Здесь и на последующих рисунках сохранена нумерация работ, принятая в ИСО/МЭК 12207. Общее число задач по данным работам равно 23.

Табл. 1 содержит задачи, реализуемые при выполнении соответствующих работ процесса заказа, и типы выходных результатов данных задач, которые должны быть документально оформлены.

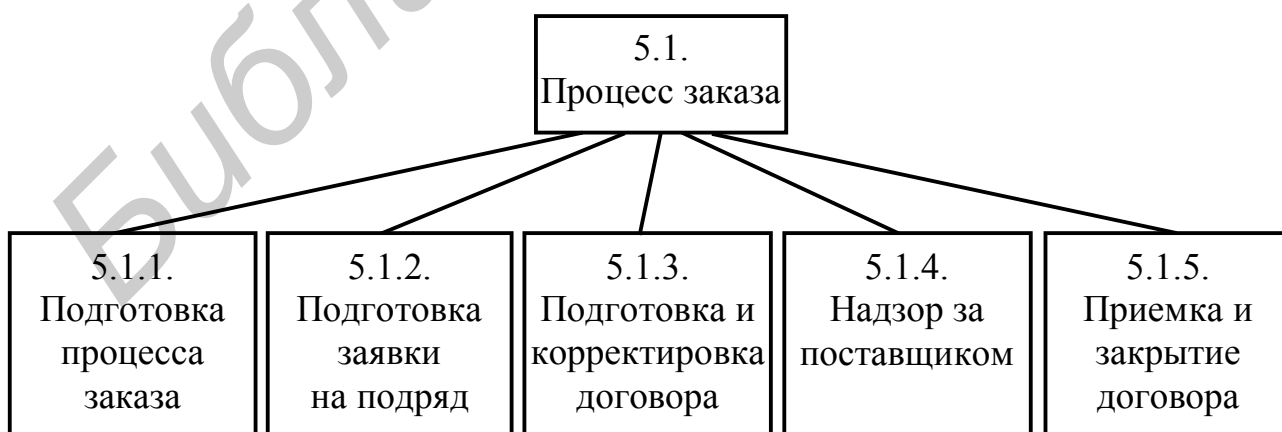


Рис. 5. Структура процесса заказа

## Задачи работ процесса заказа

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
<b>5.1.1. Задачи работы <i>Подготовка процесса заказа</i></b>		
5.1.1.1	Описание концепции или потребности в заказе	Описание (в составе результата задачи 5.1.1.8)
5.1.1.2	Анализ требований к системе	Описание (в составе результата 5.1.1.8)
5.1.1.3	Согласование проанализированных требований (выполняется, если анализ требований осуществляет не заказчик, а поставщик)	
5.1.1.4	Анализ требований к программным средствам (выполняется заказчиком или, по его поручению, поставщиком)	Описание (в составе результата 5.1.1.8)
5.1.1.5	Использование процессов разработки при выполнении задач 5.1.1.2 и 5.1.1.4	
5.1.1.6	Рассмотрение вариантов реализации заказа (покупка готового ПП, разработка собственными силами, разработка на договорной основе, модернизация существующего ПП, комбинация вариантов)	Описание (в составе результата 5.1.1.8)
5.1.1.7	Оценка условий приобретения готового программного продукта	
5.1.1.8	Документальное оформление и выполнение плана заказа	План
5.1.1.9	Определение и документальное оформление правил и условий реализации договора	Описание
<b>5.1.2. Задачи работы <i>Подготовка заявки на подряд</i></b>		
5.1.2.1	Документальное оформление требований к заказу (заявки на подряд)	Описание
5.1.2.2	Адаптация ИСО/МЭК 12207 к условиям проекта	Описание (в составе результата 5.1.2.1)
5.1.2.3	Определение контрольных точек договора	Описание (в составе результата 5.1.2.1)
5.1.2.4	Предоставление требований к заказу поставщикам для тендера (конкурса)	

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
<b>5.1.3. Задачи работы <i>Подготовка и корректировка договора</i></b>		
5.1.3.1	Определение процедуры выбора поставщика на тендерной (конкурсной) основе	Процедура
5.1.3.2	Выбор поставщика	
5.1.3.3	Окончательное решение по адаптации ИСО/МЭК 12207 к условиям проекта	Описание (в составе результата 5.1.3.4)
5.1.3.4	Подготовка и заключение договора с поставщиком	Договор
5.1.3.5	Контроль изменений, вносимых в договор, в ходе его реализации	
<b>5.1.4. Задачи работы <i>Надзор за поставщиком</i></b>		
5.1.4.1	Надзор за работами поставщика	
5.1.4.2	Взаимодействие с поставщиком для решения проблем	
<b>5.1.5. Задачи работы <i>Приемка и закрытие договора</i></b>		
5.1.5.1	Подготовка к приемке	Процедура
5.1.5.2	Проведение приемочных испытаний в соответствии с условиями задачи 5.1.1.9	Протокол (общий с результатом 5.3.13.1)
5.1.5.3	Управление конфигурацией поставленного ПП	

## 2.2.2. Процесс поставки (The Supply Process)

*Процесс поставки* определяет работы и задачи поставщика. Процесс поставки начинается с решения о подготовке предложения в ответ на заявку на подряд, присланную заказчиком, или с подписания договора с заказчиком на поставку системы, ПП или программной услуги. Затем определяются процедуры и ресурсы, необходимые для управления и обеспечения проекта, включая разработку проектных планов и их выполнение.

Процесс поставки состоит из *семи работ* (рис. 6). Общее число задач по данным работам равно 23.

Табл. 2 содержит задачи, реализуемые при выполнении соответствующих работ процесса поставки, и типы выходных результатов данных задач, которые должны быть документально оформлены.



Рис. 6. Структура процесса поставки

Таблица 2

Задачи работ процесса поставки

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
<b>5.2.1. Задачи работы <i>Подготовка процесса поставки</i></b>		
5.2.1.1	Анализ требований в заявке на подряд	
5.2.1.2	Принятие решения об участии в тендере или о подписании договора	
<b>5.2.2. Задачи работы <i>Подготовка ответа</i></b>		
5.2.2.1	Подготовка предложения в ответ на заявку о подряде, включая предложения по адаптации ИСО/МЭК 12207	Предложение
<b>5.2.3. Задачи работы <i>Подготовка договора</i></b>		
5.2.3.1	Проведение переговоров с заказчиком	
5.2.3.2	Внесение изменений в текст договора по согласованию с заказчиком	
<b>5.2.4. Задачи работы <i>Планирование</i></b>		
5.2.4.1	Анализ требований к заказу	
5.2.4.2	Определение модели жизненного цикла программных средств со структурированием в нее процессов, работ и задач ИСО/МЭК 12207	Описание (в составе результата задачи 5.2.4.5)
5.2.4.3	Определение требований к планированию управле-	Описание (в

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
	ния и обеспечения проекта, обеспечения качества ПП или программной услуги, ресурсов	составе результата 5.2.4.5)
5.2.4.4	Анализ вариантов разработки программного продукта или предоставления программной услуги	Описание (в составе результата 5.2.4.5)
5.2.4.5	Разработка и документальное оформление плана (планов) управления проектом	План
5.2.5. Задачи работы <b>Выполнение и контроль</b>		
5.2.5.1	Реализация планов управления проектом	
5.2.5.2	Разработка, проведение опытной эксплуатации и сопровождение ПП в соответствии с процессами разработки, эксплуатации и сопровождения	
5.2.5.3	Надзор за реализацией проекта, выявление и решение проблем	
5.2.5.4	Управление и контроль деятельности субподрядчиков в соответствии с процессом заказа	Процедура
5.2.5.5	Взаимодействие с верифицирующей, аттестующей или проверяющей организацией	
5.2.5.6	Взаимодействие с другими исполнителями договора	
5.2.6. Задачи работы <b>Проверка и оценка</b>		
5.2.6.1	Координация работы по проверке выполнения договора	
5.2.6.2	Участие в совещаниях, подготовке приемки ПП, приемочных испытаниях, совместных анализах и аудиторских проверках	
5.2.6.3	Выполнение верификации и аттестации ПП, программной услуги и процессов	
5.2.6.4	Предоставление заказчику отчетов об оценках, анализах, аудиторских проверках, испытаниях, решениях проблем	
5.2.6.5	Обеспечение заказчику доступа к ресурсам для проверки ПП или программной услуги	
5.2.6.6	Выполнение работ по обеспечению качества	
5.2.7. Задачи работы <b>Поставка и закрытие договора</b>		
5.2.7.1	Поставка ПП или программной услуги заказчику	



№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.2.7.2	Помощь заказчику в поддержке поставленного ПП или программной услуги	

**План управления** (см. задачу 5.2.4.5) должен содержать следующие вопросы:

- планирование организационной структуры проекта, полномочий и обязанностей каждого участника проекта;
- планирование технической среды разработки, эксплуатации и сопровождения; техническая среда включает условия проведения испытаний, оборудование, организация архивной библиотеки, средства, стандарты, процедуры, инструментарий;
- планирование структуры распределения заданий по процессам и работам жизненного цикла совместно со сметами, составом исполнителей, требуемыми материальными ресурсами, необходимыми программными средствами, графиками выполнения задач;
- планирование управления характеристиками качества создаваемого программного продукта или программной услуги;
- планирование управления безопасностью, защитой и другими критическими требованиями к программному продукту или программной услуге;
- планирование обеспечения защиты, включая правила доступа к информации на уровне каждой проектной организации;
- планирование управления критическими ситуациями (областями проекта, связанными с потенциальными техническими, финансовыми и плановыми затруднениями);
- планирование управления субподрядчиками;
- планирование обеспечения качества (см. п. 2.3.3 пособия);
- планирование верификации и аттестации (см. пп. 2.3.4, 2.3.5 пособия);
- планирование взаимоотношений с заказчиками, реализуемых совместными анализами (см. п. 2.3.6 пособия), аудиторскими проверками (см. п. 2.3.7), совещаниями, отчетами, модификациями и изменениями программного продукта, сдачей, утверждением, приемкой ПП и договорами;
- планирование взаимоотношений с пользователями, реализуемых посредством выполнения требуемых настроек, демонстраций прототипов и оценок;
- планирование подтверждения статуса поставляемой продукции (инструкции, обязательная сертификация, права собственности, использования и распространения, гарантии и лицензионные права);
- планирование средств для планирования, надзора и отчетности;
- планирование обучения персонала.

## 2.2.3. Процесс разработки (The Development Process)

**Процесс разработки** определяет работы и задачи разработчика. Данный процесс включает работы по анализу требований, проектированию, программированию, сборке, тестированию, вводу в действие и приемке программного продукта или системы.

Процесс разработки состоит из *тринадцати работ* (рис. 7). Общее число задач по данным работам равно 55.

Табл. 3 – 15 содержат задачи, реализуемые при выполнении соответствующих работ процесса разработки, и типы выходных результатов данных задач, которые должны быть документально оформлены.

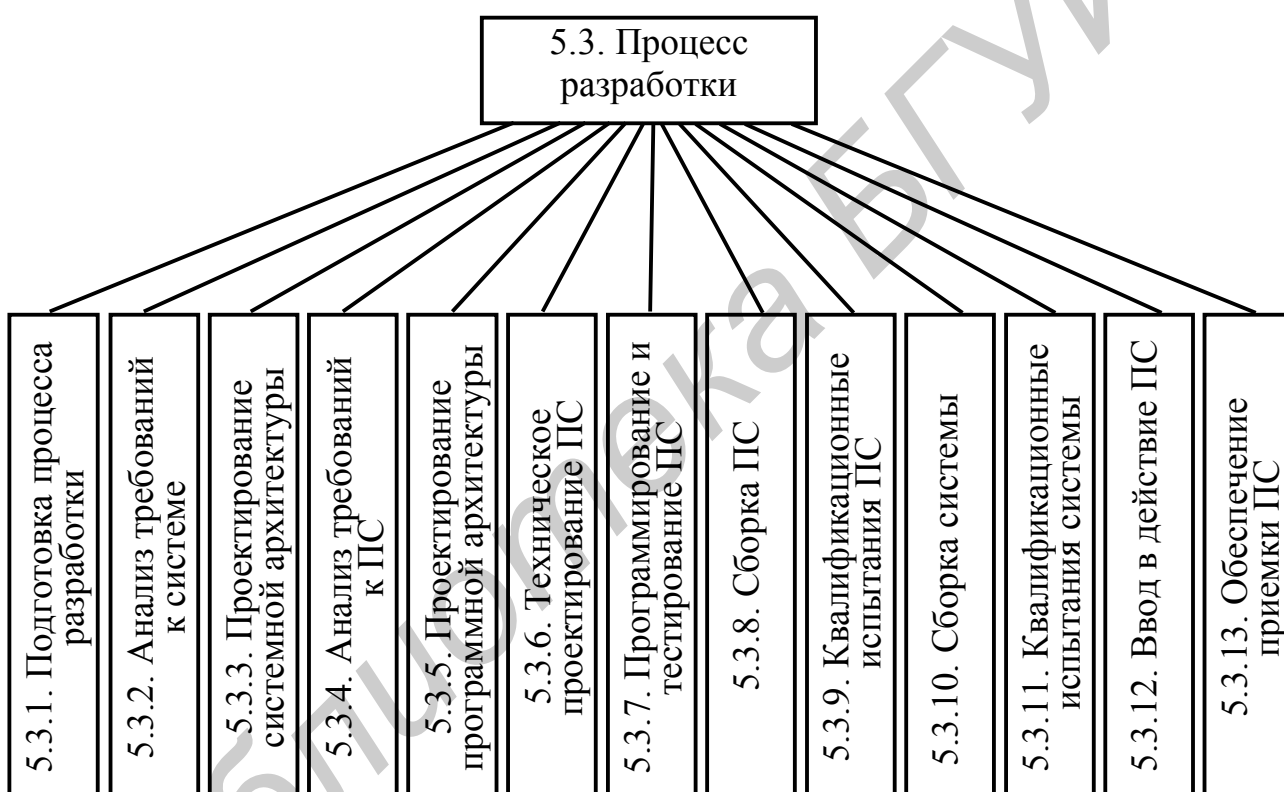


Рис. 7. Структура процесса разработки

Таблица 3

Задачи работы **Подготовка процесса разработки**

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.1.1	Определение модели жизненного цикла программных средств со структурированием в нее процессов, работ и задач ИСО/МЭК 12207 (если модель	Описание (в составе результата

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
	не определена в договоре)	задачи 5.1.3.4)
5.3.1.2	Документальное оформление выходных результатов, управление конфигурацией выходных результатов, решение возникающих проблем, выполнение вспомогательных процессов в соответствии с процессами группы 6 (рис. 13, подразд. 2.3)	Протоколы и отчеты в составе результатов соответствующих работ
5.3.1.3	Выбор и адаптация стандартов, методов, инструментариев, языков программирования	Описание (в составе результата 5.1.3.4)
5.3.1.4	Разработка и выполнение планов проведения работ процесса разработки	План
5.3.1.5	Поставка всех комплектующих изделий	

Таблица 4

Задачи работы *Анализ требований к системе*

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.2.1	Анализ области применения системы и определение требований к ней (рис. 8)	Описание
5.3.2.2	Оценка требований к системе по критериям (см. рис. 8)	Отчет

Таблица 5

Задачи работы *Проектирование системной архитектуры*

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.3.1	Определение общей архитектуры системы, распределение требований к ней между объектами технических и программных средств архитектуры и ручными операциями и дальнейшее уточнение требований	Описание
5.3.3.2	Оценка архитектуры системы и требований к объектам архитектуры по критериям	Отчет

При оценке архитектуры системы и требований к ее объектам (см. задачу 5.3.3.2 в табл. 5) должны быть проанализированы следующие критерии:

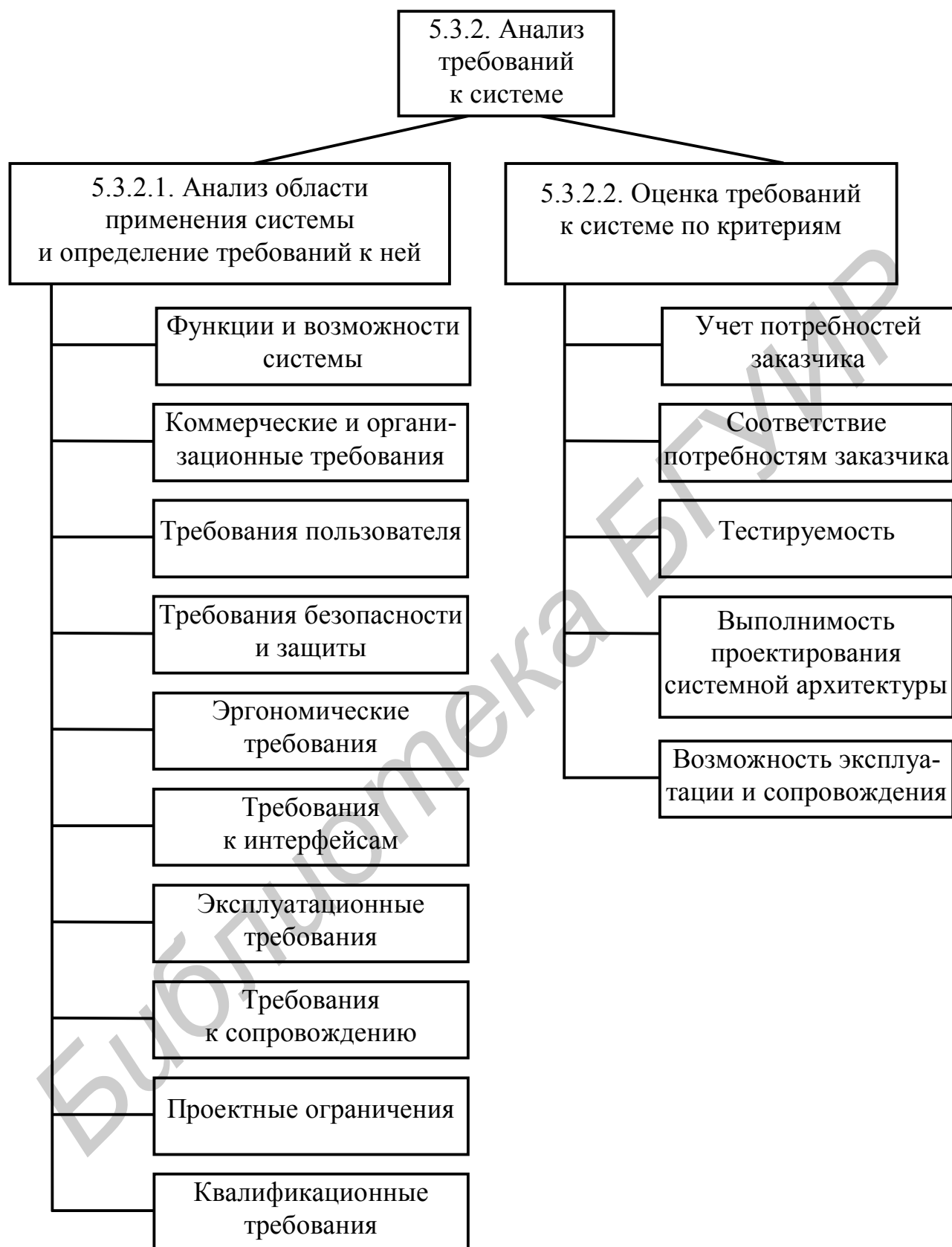


Рис. 8. Состав требований к системе и критерии их оценки

- учет требований к системе;
- соответствие требованиям к системе;
- соответствие используемых стандартов и методов проектирования;
- возможность программных объектов архитектуры выполнять установленные для них требования;
- возможность эксплуатации и сопровождения.

Таблица 6

Задачи работы *Анализ требований к программным средствам*

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.4.1	Определение требований к ПС (рис. 9)	Описание
5.3.4.2	Оценка требований к ПС по критериям (см. рис. 9)	Отчет
5.3.4.3	Проведение совместных анализов в соответствии с процессом совместного анализа	Протокол

Таблица 7

Задачи работы *Проектирование программной архитектуры*

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.5.1	Преобразование требований к программному объекту в его архитектуру, распределение требований к программному объекту между его компонентами и дальнейшее уточнение требований (разработка эскизного проекта)	Описание
5.3.5.2	Разработка эскизного проекта интерфейсов программного объекта и его компонентов	Описание
5.3.5.3	Разработка эскизного проекта базы данных	Описание
5.3.5.4	Разработка предварительных версий документации пользователя	Руководство
5.3.5.5	Разработка предварительных требований к испытаниям (тестированию) программного объекта и графика сборки программного продукта	Описание, план
5.3.5.6	Оценка архитектуры программного объекта и эскизных проектов интерфейсов и базы данных по критериям	Отчет
5.3.5.7	Проведение совместных анализов в соответствии с процессом совместного анализа	



Рис. 9. Состав требований к программным средствам и критерии их оценки

При оценке архитектуры программного объекта и эскизных проектов интерфейсов и базы данных (см. задачу 5.3.5.6) должны быть учтены критерии:

- учет требований к программному объекту;
- внешняя согласованность с требованиями к программному объекту;
- внутренняя согласованность между компонентами объекта;
- соответствие методов проектирования и используемых стандартов;
- возможность технического проектирования;
- возможность эксплуатации и сопровождения.

Таблица 8

Задачи работы *Техническое проектирование программных средств*

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.6.1	Разработка технического проекта для компонентов программного объекта (представление их в виде набора программных модулей), распределение технических требований к компонентам между программными модулями и дальнейшее уточнение требований	Описание
5.3.6.2	Разработка технического проекта интерфейсов программного объекта, его компонентов и модулей	Описание
5.3.6.3	Разработка технического проекта базы данных	Описание
5.3.6.4	Уточнение документации пользователя	Руководство
5.3.6.5	Разработка требований к испытаниям и программе испытаний программных модулей	Описание, план
5.3.6.6	Уточнение общих требований к испытанию (тестированию) и программе сборки программных средств	Описание, план
5.3.6.7	Оценка технического проекта и требований к тестированию по критериям	Отчет
5.3.6.8	Проведение совместных анализов в соответствии с процессом совместного анализа	

При оценке технического проекта и требований к тестированию (см. задачу 5.3.6.7) должны быть учтены следующие критерии:

- учет требований к программному объекту;
- внешнее соответствие спроектированной архитектуре;
- внутренняя согласованность между компонентами программного объекта и программными модулями;
- соответствие методов проектирования и используемых стандартов;
- возможность тестирования;
- возможность эксплуатации и сопровождения.

Работы по проектированию программных средств (см. табл. 4 – табл. 8) поясняет рис. 10.

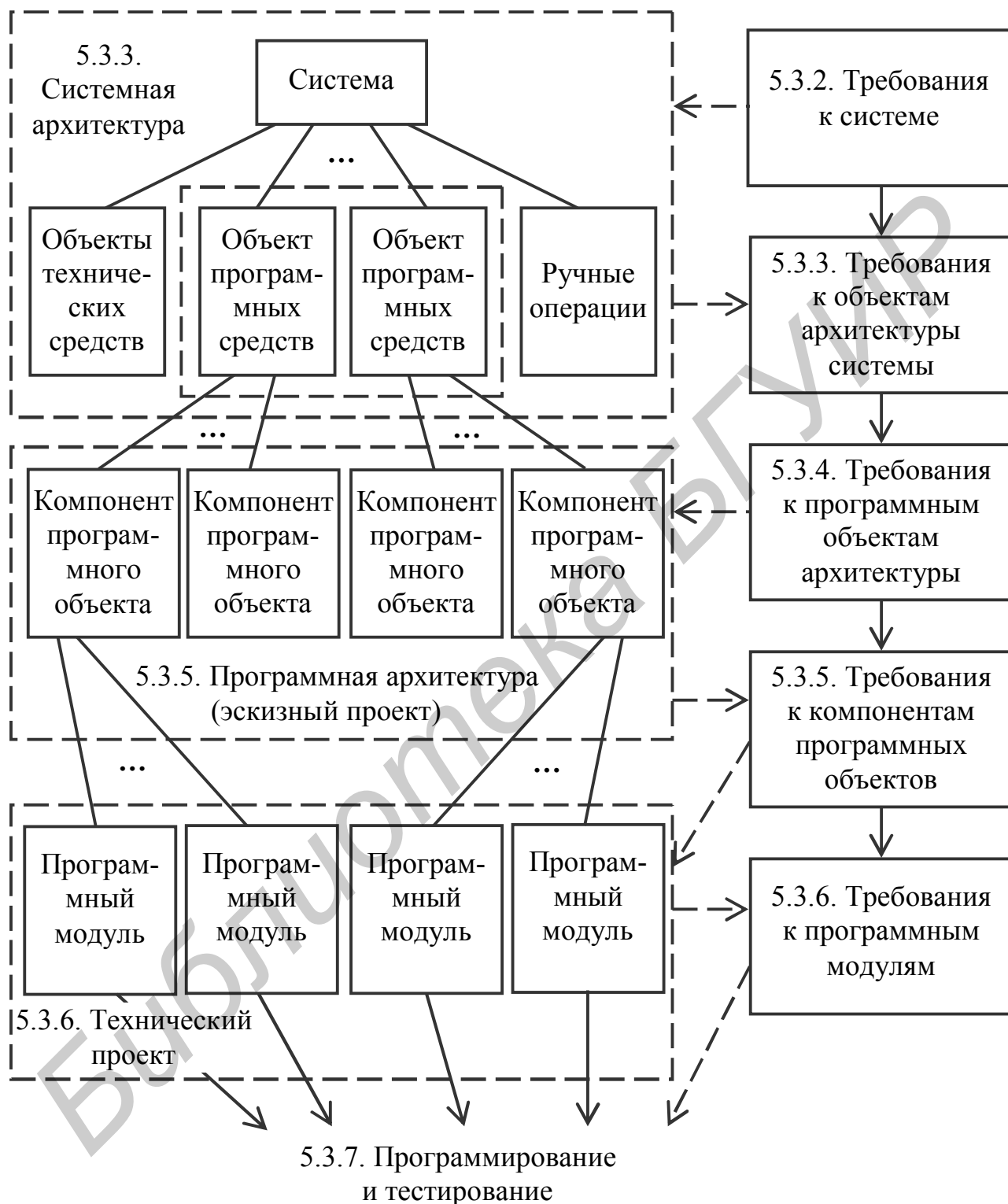


Рис. 10. Структура результатов работ по проектированию программных средств



На данном рисунке отражена структура основных результатов работ процесса разработки, связанных с проектированием ПС, и взаимосвязь данных результатов.

Таблица 9

Задачи работы *Программирование и тестирование программных средств*

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.7.1	Программирование программных модулей, базы данных, разработка процедур и данных для их испытаний (тестирования)	Программный продукт, описание, процедура
5.3.7.2	Тестирование программных модулей и базы данных	Протокол
5.3.7.3	Уточнение документации пользователя	Руководство
5.3.7.4	Уточнение общих требований к тестированию и программы сборки программного средства	Описание, план
5.3.7.5	Оценка результатов программирования и тестирования по критериям	Отчет

При оценке результатов программирования и тестирования (см. задачу 5.3.7.5) должны быть учтены следующие критерии:

- учет требований к программному объекту и проекту объекта в целом;
- внешнее соответствие требованиям и проекту программного объекта;
- внутреннее соответствие между требованиями к программным модулям;
- тестовое покрытие всех модулей;
- соответствие методов программирования и используемых стандартов;
- возможность сборки и тестирования;
- возможность эксплуатации и сопровождения.

Таблица 10

Задачи работы *Сборка программных средств*

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.8.1	Разработка плана сборки для объединения программных модулей и компонентов в программный объект	План
5.3.8.2	Сборка программных модулей и компонентов и тестирование результатов сборки	Описание, протокол

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.8.3	Уточнение документации пользователя	Руководство
5.3.8.4	Разработка плана и процедуры квалификационных испытаний программных средств	План, процедура
5.3.8.5	Оценка планов и результатов сборки и тестирования, а также документации пользователя по критериям	Отчет
5.3.8.6	Проведение совместного анализа в соответствии с процессом совместного анализа (см. п. 2.3.6 пособия)	

При оценке планов и результатов сборки и тестирования, а также документации пользователя (см. задачу 5.3.8.5) должны быть учтены следующие критерии:

- учет требований к системе;
- внешнее соответствие требованиям к системе;
- внутренняя согласованность между программными объектами;
- тестовое покрытие требований к программному объекту;
- соответствие используемых стандартов и методов испытаний;
- соответствие ожидаемым результатам;
- выполнимость квалификационного испытания программного объекта;
- возможность эксплуатации и сопровождения.

Таблица 11

Задачи работы **Квалификационные испытания программных средств**

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.9.1	Проведение квалификационных испытаний (тестирования) программных средств	Протокол
5.3.9.2	Уточнение документации пользователя	Руководство
5.3.9.3	Оценка результатов испытаний и документации пользователя по критериям	Отчет
5.3.9.4	Обеспечение проведения аудиторской проверки в соответствии с процессом аудита (см. п. 2.3.7 пособия)	Протокол

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.9.5	Доработка и подготовка программного продукта к последующим работам, определение состояния конфигурации (базовой линии) проекта	Протокол

При оценке результатов испытаний и документации пользователя (см. задачу 5.3.9.3) должны быть учтены следующие критерии:

- тестовое покрытие требований к программному объекту;
- соответствие ожидаемым результатам;
- возможность сборки и тестирования системы;
- возможность эксплуатации и сопровождения.

Таблица 12

Задачи работы *Сборка системы*

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.10.1	Сборка объектов программной и технической конфигурации, ручных операций, других систем в единую систему, испытания собранной системы	Описание, протокол
5.3.10.2	Разработка плана и процедуры квалификационных испытаний системы	План, процедура
5.3.10.3	Оценка собранной системы по критериям	Отчет

При оценке собранной системы (см. задачу 5.3.10.3) должны быть учтены следующие критерии:

- тестовое покрытие требований к системе;
- соответствие методов тестирования и используемых стандартов;
- соответствие ожидаемым результатам;
- возможность квалификационных испытаний системы;
- возможность эксплуатации и сопровождения.

Таблица 13

Задачи работы *Квалификационные испытания системы*

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.11.1	Проведение квалификационных испытаний системы	Описание, протокол

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.11.2	Оценка системы по критериям	Отчет
5.3.11.3	Обеспечение проведения аудиторской проверки в соответствии с процессом аудита	Протокол
5.3.11.4	Доработка и подготовка программного продукта к последующим работам, определение состояния конфигурации (базовой линии) проекта	Протокол

При *оценке системы* (см. задачу 5.3.11.2) должны быть учтены следующие критерии:

- тестовое покрытие требований к системе;
- соответствие ожидаемым результатам;
- возможность эксплуатации и сопровождения.

Таблица 14

Задачи работы ***Ввод в действие программных средств***

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.12.1	Разработка плана ввода в действие программного продукта в среде эксплуатации	План
5.3.12.2	Ввод в действие программного продукта в соответствии с планом	Протокол

Таблица 15

Задачи работы ***Обеспечение приемки программных средств***

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.3.13.1	Обеспечение проведения заказчиком приемочных испытаний	Процедура протокол
5.3.13.2	Укомплектование и поставка программного продукта заказчику на условиях договора	Программный продукт
5.3.13.3	Обеспечение обучения и поддержка персонала заказчика на условиях договора	

## 2.2.4. Процесс эксплуатации (The Operation Process)

*Процесс эксплуатации* определяет работы и задачи оператора. Данный процесс включает эксплуатацию программного продукта и поддержку пользователей в процессе эксплуатации. Процесс эксплуатации состоит из *четырёх работ* (рис. 11). Общее число задач по данным работам равно 9.

Табл. 16 содержит задачи, реализуемые при выполнении соответствующих работ процесса эксплуатации, и типы выходных результатов данных задач, которые должны быть документально оформлены.

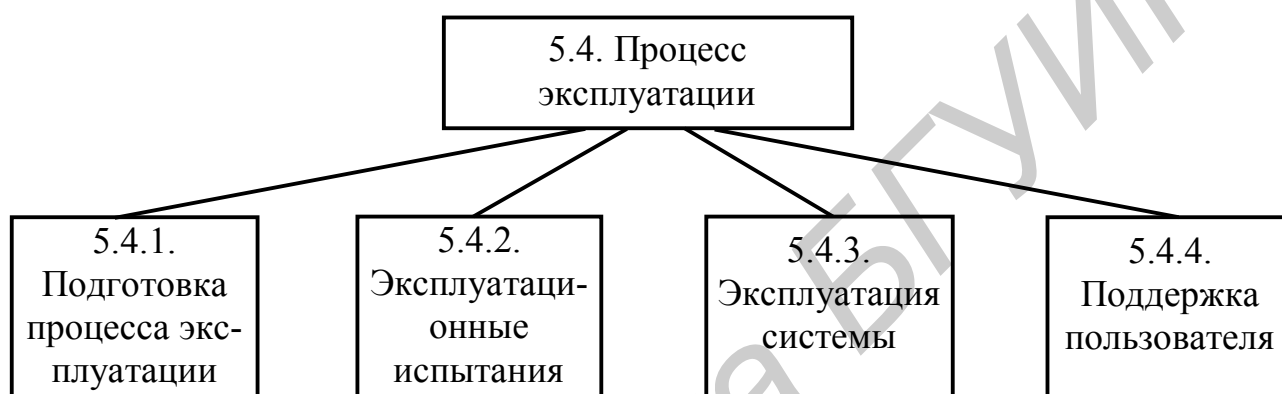


Рис. 11. Структура процесса эксплуатации

Таблица 16

Задачи работ процесса эксплуатации

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
<b>5.4.1. Задачи работы <i>Подготовка процесса эксплуатации</i></b>		
5.4.1.1	Разработка плана эксплуатации и определение набора стандартов по эксплуатации	План
5.4.1.2	Установка процедур документирования и решения проблем	Процедура
5.4.1.3	Установка процедур для тестирования ПП в эксплуатационной среде, ввода сообщений о проблеме в процесс сопровождения и ввода ПП в эксплуатацию	Процедура
<b>5.4.2. Задачи работы <i>Эксплуатационные испытания</i></b>		
5.4.2.1	Проведение эксплуатационных испытаний и ввод программных продуктов в промышленную эксплуатацию	Протокол

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
5.4.2.2	Обеспечение инициализации и эксплуатации программ и базы данных в соответствии с планом эксплуатации	
5.4.3. Задачи работы <i>Эксплуатация системы</i>		
5.4.3.1	Эксплуатация в установленной среде в соответствии с документацией пользователя	
5.4.4. Задачи работы <i>Поддержка пользователя</i>		
5.4.4.1	Помощь и консультации пользователям в установленном порядке	Протокол
5.4.4.2	Передача запросов пользователя в процесс сопровождения и контроль их решения	Протокол
5.4.4.3	Обеспечение временного решения проблем	

## 2.2.5. Процесс сопровождения (The Maintenance Process)

*Процесс сопровождения* определяет работы и задачи персонала сопровождения и реализуется при модификациях программного продукта. Цель процесса – изменение существующего ПП при сохранении его целостности. Процесс охватывает вопросы переносимости и снятия ПП с эксплуатации.

Процесс сопровождения состоит из *шести работ* (рис. 12).



Рис. 12. Структура процесса сопровождения

Общее число задач по данным работам равно 24.

Табл. 17 содержит задачи, реализуемые при выполнении соответствующих работ процесса сопровождения, и типы выходных результатов данных задач, которые должны быть документально оформлены.

Таблица 17

Задачи работ процесса сопровождения

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
<b>5.5.1. Задачи работы <i>Подготовка процесса сопровождения</i></b>		
5.5.1.1	Разработка планов и процедур для проведения работ по сопровождению	План, процедура
5.5.1.2	Определение процедур для документирования возникающих проблем и организации связи с процессом решения проблем (см. п. 2.3.8 пособия)	Процедура
5.5.1.3	Реализация процесса управления конфигурацией (см. п. 2.3.2 пособия) для управления изменениями существующей системы	
<b>5.5.2. Задачи работы <i>Анализ проблем и изменений</i></b>		
5.5.2.1	Анализ сообщений о проблеме	
5.5.2.2	Дублирование или верификация проблемы	
5.5.2.3	Разработка вариантов реализации изменения	
5.5.2.4	Документальное оформление сообщения о проблеме и вариантов ее решения	Протокол, отчет
5.5.2.5	Согласование выбранного варианта изменения в соответствии с договором	
<b>5.5.3. Задачи работы <i>Внесение изменений</i></b>		
5.5.3.1	Определение документов и программных модулей, требующих изменения	Протокол
5.5.3.2	Реализация изменений с использованием процесса разработки, разработка процедур испытаний и оценки результатов испытаний	Процедура, протокол, отчет
<b>5.5.4. Задачи работы <i>Проверка и приемка при сопровождении</i></b>		
5.5.4.1	Совместная проверка работоспособности измененной системы	
5.5.4.2	Приемка внесенного изменения	Протокол
<b>5.5.5. Задачи работы <i>Перенос</i></b>		
5.5.5.1	Обеспечение соответствия измененного при пере-	

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
	носе в новую эксплуатационную среду программного продукта или данных стандарту <i>ИСО/МЭК 12207</i>	
5.5.5.2	Разработка и выполнение плана переноса программного продукта	План
5.5.5.3	Уведомление пользователей о планах и работах по переносу программного продукта	Отчет
5.5.5.4	Обеспечение параллельной эксплуатации в прежней и новой среде (при необходимости)	
5.5.5.5	Уведомление заинтересованных сторон о выполненном переносе, передача в архив документации по прежней среде	
5.5.5.6	Итоговый анализ влияния переноса на эксплуатацию системы и рассылка результатов анализа заинтересованным сторонам	Отчет
5.5.5.7	Обеспечение доступности информации и документации, связанных с прежней средой	
<b>5.5.6. Задачи работы <i>Снятие с эксплуатации</i></b>		
5.5.6.1	Разработка плана снятия с эксплуатации	План
5.5.6.2	Уведомление пользователей о планах и работах по снятию с эксплуатации	Отчет
5.5.6.3	Проведение параллельной эксплуатации прежнего и нового программных продуктов	
5.5.6.4	Уведомление заинтересованных сторон о снятии ПП с эксплуатации и передача документации о нем в архив	
5.5.6.5	Обеспечение доступности информации по снятому с эксплуатации ПП	

## **2.3. Вспомогательные процессы жизненного цикла**

Вспомогательные процессы жизненного цикла ПС и систем представлены на рис. 13. На данном рисунке сохранена нумерация процессов, принятая в стандарте *ИСО/МЭК 12207*.



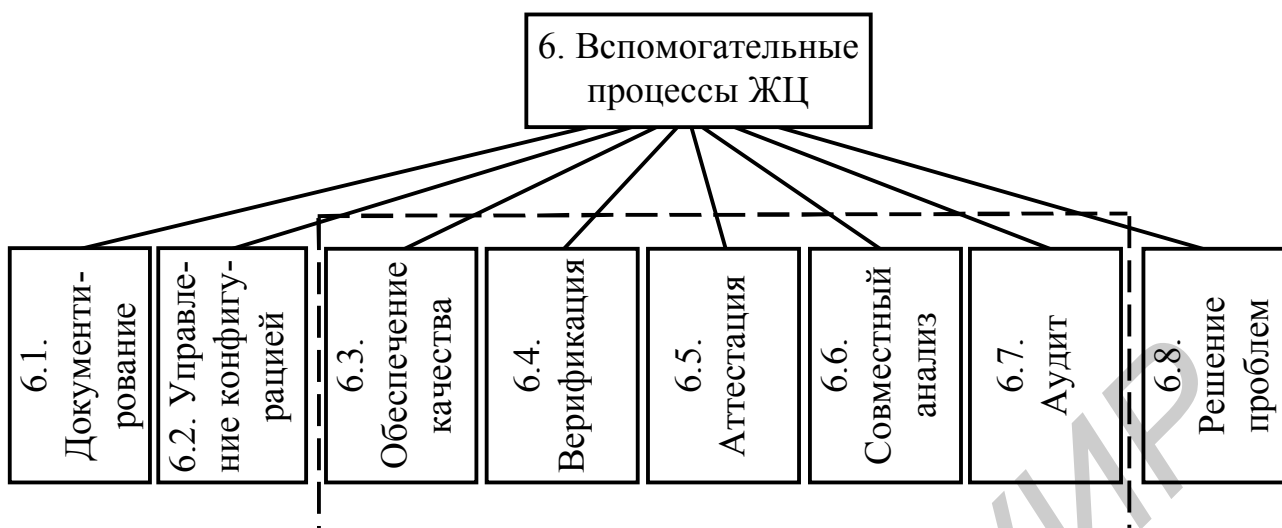


Рис. 13. Вспомогательные процессы жизненного цикла программных средств и систем

### 2.3.1. Процесс документирования (The Documentation Process)

*Процесс документирования* является процессом формализованного описания информации, созданной в процессе или работе жизненного цикла. Он включает планирование, проектирование, разработку, выпуск, редактирование, распространение и сопровождение документов по программному продукту.

Процесс документирования состоит из *четырёх работ* (рис. 14). Общее число задач по данным работам равно 7.

Табл. 18 содержит задачи, реализуемые при выполнении соответствующих работ процесса документирования, и типы выходных результатов данных задач, которые должны быть документально оформлены.

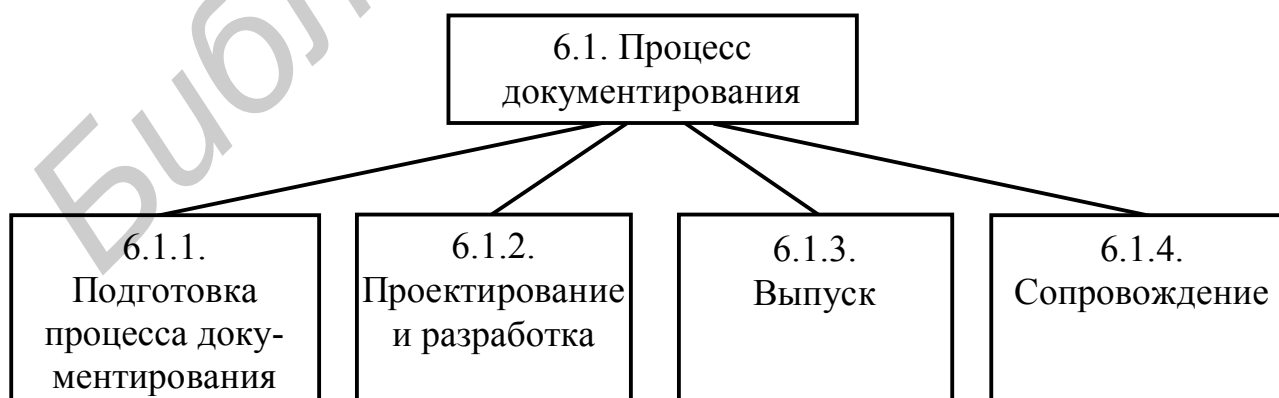


Рис. 14. Структура процесса документирования

## Задачи работ процесса документирования

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
6.1.1. Задачи работы <b>Подготовка процесса документирования</b>		
6.1.1.1	Разработка и реализация плана обозначения документов, выпускаемых в процессах ЖЦ ПП	План
6.1.2. Задачи работы <b>Проектирование и разработка</b>		
6.1.2.1	Проектирование документов согласно стандартам на документацию	Описание
6.1.2.2	Подтверждение источника и соответствия исходных материалов для документов	
6.1.2.3	Проверка и редактирование документов согласно стандартам, утверждение компетентными лицами	
6.1.3. Задачи работы <b>Выпуск</b>		
6.1.3.1	Издание и распространение документов в соответствии с планом	
6.1.3.2	Управление документированием в соответствии с процессом управления конфигурацией	
6.1.4. Задачи работы <b>Сопровождение</b>		
6.1.4.1	Внесение изменений в документацию согласно процессам сопровождения и управления конфигурацией	

В *плане обозначения документов* (см. задачу 6.1.1.1) должны быть определены:

- заголовок или наименование;
- назначение;
- пользователи документа;
- процедуры и обязанности по подготовке исходных материалов, разработке, проверке, изменению, утверждению, выпуску, хранению, распространению, сопровождению и управлению конфигурацией документов.

## 2.3.2. Процесс управления конфигурацией (The Configuration Management Process)

*Процесс управления конфигурацией* является процессом применения административных и технических процедур на всем протяжении ЖЦ ПС для определения состояния (базовой линии) программных объектов в системе, управления их изменениями и выпуском.

Данный процесс состоит из *шести работ* (рис. 15). Общее число задач по данным работам равно 6.

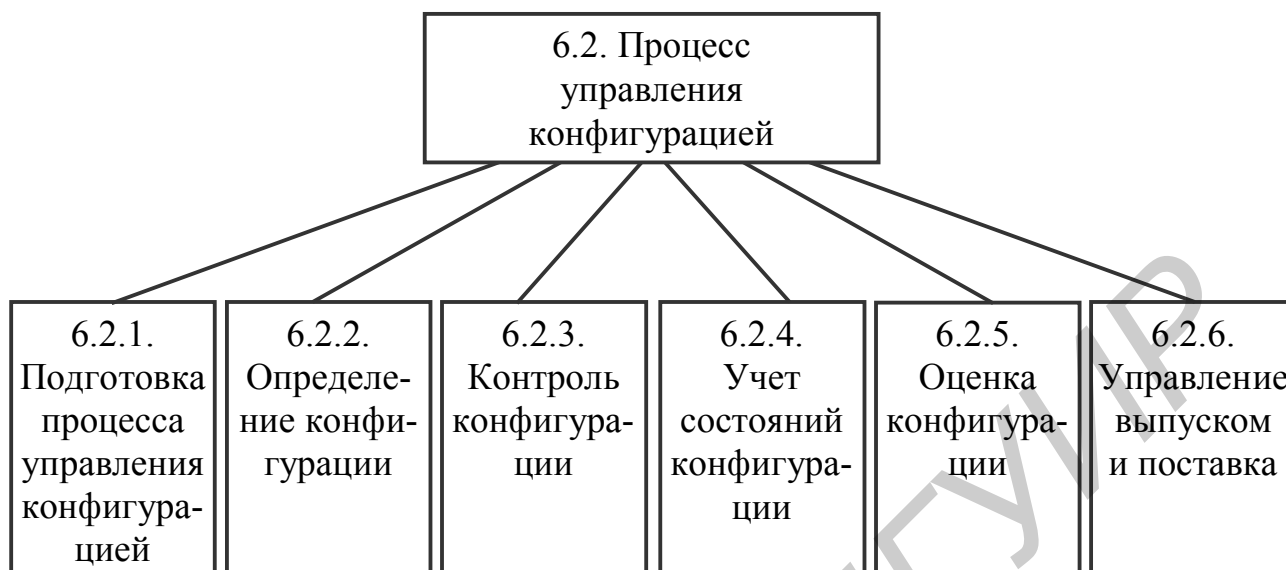


Рис. 15. Структура процесса управления конфигурацией

Табл. 19 содержит задачи, реализуемые при выполнении соответствующих работ процесса управления конфигурацией, и типы выходных результатов данных задач, которые должны быть документально оформлены.

Таблица 19

Задачи работ процесса управления конфигурацией

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
<b>6.2.1. Задачи работы <i>Подготовка процесса управления конфигурацией</i></b>		
6.2.1.1	Разработка плана управления конфигурацией	План
<b>6.2.2. Задачи работы <i>Определение конфигурации</i></b>		
6.2.2.1	Определение схемы обозначения программных объектов и их версий (объектов программной конфигурации) и документации, в которой фиксируется состояние их конфигурации	Описание
<b>6.2.3. Задачи работы <i>Контроль конфигурации</i></b>		
6.2.3.1	Регистрация заявок на внесение изменений; анализ и оценка изменений; принятие или непринятие заявки; реализация, верификация и выпуск измененного программного объекта; обеспечение аудиторских проверок изменений	

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
6.2.4. Задачи работы <i>Учет состояний конфигурации</i>		
6.2.4.1	Подготовка протоколов управления конфигурацией и отчетов о состоянии контролируемых программных объектов	Протокол, отчет
6.2.5. Задачи работы <i>Оценка конфигурации</i>		
6.2.5.1	Определение и обеспечение функциональной законченности и физической завершенности программных объектов	Протокол, отчет
6.2.6. Задачи работы <i>Управление выпуском и поставка</i>		
6.2.6.1	Контроль выпуска и поставки программных продуктов и документации	

### 2.3.3. Процесс обеспечения качества (The Quality Assurance Process)

*Процесс обеспечения качества* является процессом обеспечения гарантий того, что программные продукты и процессы в жизненном цикле проекта соответствуют требованиям и планам. Данный процесс должен быть независимым от субъектов, участвующих в проекте. Это позволяет достичь объективности процесса. При обеспечении качества могут использоваться результаты процессов верификации, аттестации, совместного анализа, аудита и решения проблем (см. пп. 2.3.4, 2.3.5, 2.3.6, 2.3.7, 2.3.8 пособия).

Процесс обеспечения качества состоит из *четырёх работ* (рис. 16). Общее число задач по данным работам равно 16.

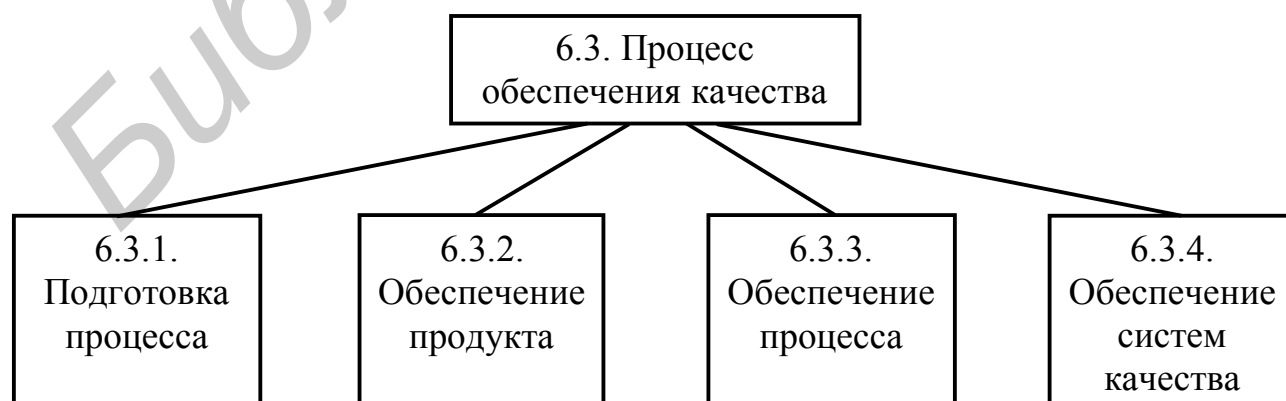


Рис. 16. Структура процесса обеспечения качества

Табл. 20 содержит задачи, реализуемые при выполнении соответствующих работ процесса обеспечения качества, и типы выходных результатов данных задач, которые должны быть документально оформлены.

Таблица 20

Задачи работ процесса обеспечения качества

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
<b>6.3.1. Задачи работы <i>Подготовка процесса обеспечения качества</i></b>		
6.3.1.1	Адаптация процесса обеспечения качества к условиям конкретного проекта, определение целей процесса обеспечения качества	Описание (в составе результата задачи 6.3.1.3)
6.3.1.2	Координация с процессами верификации, аттестации, совместного анализа и аудита (см. пп. 2.3.4, 2.3.5, 2.3.6, 2.3.7 пособия)	
6.3.1.3	Разработка и реализация плана обеспечения качества	План
6.3.1.4	Выполнение запланированных и традиционных работ и задач по обеспечению качества, при необходимости организация связи с процессом решения проблем (см. п. 2.3.8 пособия)	Отчет, протокол
6.3.1.5	Предоставление отчетов о работах и задачах по обеспечению качества заказчику согласно договору	
6.3.1.6	Обеспечение организационной независимости лиц, реализующих работы и задачи по обеспечению качества	
<b>6.3.2. Задачи работы <i>Обеспечение продукта</i></b>		
6.3.2.1	Обеспечение соответствия условиям договора и выполнения всех планов	
6.3.2.2	Обеспечение соответствия программных продуктов и документации условиям договора и планам	
6.3.2.3	Обеспечение соответствия поставляемых программных продуктов требованиям, установленным в договоре	
<b>6.3.3. Задачи работы <i>Обеспечение процесса</i></b>		
6.3.3.1	Обеспечение соответствия процессов жизненного цикла программных средств условиям договора и утвержденным планам	

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
6.3.3.2	Обеспечение соответствия технологий программирования, условий разработки, условий испытаний и архивных библиотек условиям договора	
6.3.3.3	Обеспечение соответствия программных продуктов, разработанных субподрядчиком, требованиям основного договора	
6.3.3.4	Обеспечение взаимной поддержки заказчика и других участников договора в соответствии с условиями договора и планами	
6.3.3.5	Обеспечение соответствия характеристик программных продуктов и процессов установленным стандартам и процедурам	
6.3.3.6	Обеспечение соответствия и обучения исполнителей проекта	
6.3.4. Задачи работы <b>Обеспечение систем качества</b>		
6.3.4.1	Обеспечение проведения дополнительных работ по управлению качеством в соответствии с разделами <i>ГОСТ Р ИСО 9001–96</i> [10], указанными в договоре	Протокол

## 2.3.4. Процесс верификации (The Verification Process)

Стандарт *ИСО/МЭК 12207* дает такое определение процесса верификации.

**Процесс верификации** является процессом определения того, что программные продукты функционируют в полном соответствии с требованиями и условиями, реализованными в предшествующих работах.

Данное определение является не совсем понятным для лиц, некомпетентных в области верификации. Кроме того, оно сужает применимость верификации только до программных продуктов. Для пояснения данного определения следует рассмотреть само понятие верификации. Термин *верификация (verification)* обозначает подтверждение с помощью экспертизы и представления объективных доказательств того, что конкретные требования полностью реализованы. В процессе разработки верификация связана с экспертизой результатов данной работы с целью определения их соответствия установленным на входе данной работы требованиям [47, 17]. Таким образом, верификация может применяться не только к программным продуктам, но и к любым другим результатам работы. Например, верификации могут подвергаться требования, системная

и программная архитектура, документация, методы, планы и т.п.

В этой связи более понятным и близким к истине является следующее определение процесса верификации. **Процесс верификации** – это процесс определения того, что результаты работы соответствуют требованиям или условиям, установленным на входе данной работы.

Процесс верификации может включать анализ, проверку и тестирование. Объектами анализа и проверки могут являться, например, документация и исходные тексты программных модулей. Объектами тестирования могут являться исполнимые коды программных модулей, компонентов, промежуточных и конечного программных продуктов.

Данный процесс называется **процессом независимой верификации**, если организация–исполнитель не зависит от поставщика, разработчика, оператора или персонала сопровождения.

Процесс верификации состоит из *двух работ* (рис. 17). Общее число задач по данным работам равно 13.



Рис. 17. Структура процесса верификации

Табл. 21 содержит задачи, реализуемые при выполнении соответствующих работ процесса верификации, и типы выходных результатов данных задач, которые должны быть документально оформлены.

Таблица 21

Задачи работ процесса верификации

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
<b>6.4.1. Задачи работы <i>Подготовка процесса верификации</i></b>		
6.4.1.1	Определение необходимости в проекте работ по верификации и степени их организационной независимости, анализ критичности проектных требо-	Описание (в составе результата

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
	ваний	задачи 6.4.1.5)
6.4.1.2	Установка процесса верификации (при необходимости)	Описание (в составе результата 6.4.1.5)
6.4.1.3	Выбор соответствующей независимой квалифицированной организации (при необходимости)	Описание (в составе результата 6.4.1.5)
6.4.1.4	Определение верифицируемых работ и продуктов, выбор работ и задач верификации	Описание (в составе результата 6.4.1.5)
6.4.1.5	Разработка плана верификации	План
6.4.1.6	Реализация плана проведения верификации, обеспечение доступности его результатов заказчику, при необходимости организация связи с процессом решения проблем (см. п. 2.3.8 пособия)	Отчет, протокол
<b>6.4.2. Задачи работы <i>Верификация</i></b>		
6.4.2.1	Верификация договора по критериям	
6.4.2.2	Верификация процесса по критериям	
6.4.2.3	Верификация требований по критериям	
6.4.2.4	Верификация результатов проектирования по критериям	
6.4.2.5	Верификация исходных текстов программных модулей по критериям	
6.4.2.6	Верификация сборки по критериям	
6.4.2.7	Верификация документации по критериям	

*Договор* должен быть верифицирован (см. задачу 6.4.2.1) по следующим критериям:

- возможность поставщика удовлетворять установленным требованиям;
- непротиворечивость требований и охват ими потребностей пользователя;
- наличие соответствующих процедур для внесения изменений в установленные требования и для решения проблем;
- наличие процедур по взаимодействию и кооперации между участниками договора;
- наличие критериев и процедур, предусмотренных в соответствии с ус-



тановленными требованиями.

**Процесс** должен быть верифицирован (см. задачу 6.4.2.2) по следующим критериям:

- соответствие и своевременность установления требований к планированию проекта;
- пригодность, реализуемость, выполнимость в соответствии с планом и условиями договора выбранных для проекта процессов;
- применимость стандартов, процедур и условий к процессам проекта;
- укомплектованность и обученность персонала в соответствии с условиями договора.

**Требования** должны быть верифицированы (см. задачу 6.4.2.3) по следующим критериям:

- непротиворечивость, выполнимость и тестируемость требований к системе;
- распределение требований к системе между объектами технических и программных средств и ручных операций в соответствии с критериями проектирования;
- непротиворечивость, выполнимость, тестируемость и точность отражения требований к системе в требованиях к программным средствам;
- правильность (подтвержденная соответствующими методами) критических требований к программным средствам, в том числе по безопасности и защите.

**Результаты проектирования** должны быть верифицированы (см. задачу 6.4.2.4) по следующим критериям:

- правильность, соответствие установленным требованиям и учет этих требований;
- реализация соответствующей последовательности событий, исходных данных, выходных результатов, интерфейсов, логики; соответствие временным ограничениям и ограничениям размера; обнаружение, локализация и устранение ошибок;
- возможность дальнейшего использования с учетом требований;
- правильность, подтвержденная соответствующими методами, реализации требований безопасности, защиты и других критических требований.

**Исходные тексты программных модулей** должны быть верифицированы (см. задачу 6.4.2.5) по следующим критериям:

- трассируемость с результатами проектирования и требованиями; тестируемость, правильность и соответствие требованиям и стандартам программирования;
- реализация соответствующей последовательности событий, соответствующих интерфейсов, правильных данных и логики управления; завершенность; соответствие временным ограничениям и ограничениям размера; обнаружение, локализация и устранение ошибок;
- соответствие результатам проектирования и требованиям;

- возможность дальнейшего использования с учетом результатов тестирования и требований;
- правильность, подтвержденная соответствующими методами, реализации требований безопасности, защиты и других критических требований.

**Сборка** должна быть верифицирована (см. задачу 6.4.2.6) по следующим критериям:

- полнота и правильность сборки программных компонентов и модулей каждого программного объекта в соответствующий программный объект;
- полнота и правильность сборки технических и программных объектов и ручных операций в систему;
- выполнение задач сборки в соответствии с планом сборки.

**Документация** должна быть верифицирована (см. задачу 6.4.2.7) по следующим критериям:

- соответствие, полнота и непротиворечивость документации;
- своевременность подготовки документации;
- соблюдение установленных процедур управления конфигурацией документов.

### 2.3.5. Процесс аттестации (The Validation Process)

В стандарте *ИСО/МЭК 12207* дано такое определение процесса аттестации.

**Процесс аттестации** является процессом определения полноты соответствия установленных требований, созданной системы или программного продукта их функциональному назначению.

Чтобы пояснить это определение, следует рассмотреть понятие аттестации.

Термин *аттестация (validation)* обозначает подтверждение экспертизой и представлением объективных доказательств того, что конкретные требования к конкретным объектам полностью реализованы. В процессе разработки аттестация связана с экспертизой продукта в целях определения его соответствия потребностям пользователя (то есть *исходным требованиям к проекту*) [47, 17].

Аттестации (проверке на соответствие исходным требованиям) могут подвергаться любые промежуточные продукты процесса разработки. Аттестация может проводиться на начальных этапах работы и как часть работы по обеспечению приемки ПС.

Данный процесс называется процессом **независимой аттестации**, если организация–исполнитель не зависит от поставщика, разработчика, оператора или персонала сопровождения.

Процесс аттестации состоит из *двух работ* (рис. 18). Общее число задач по данным работам равно 10.

Табл. 22 содержит задачи, реализуемые при выполнении соответствующих работ процесса аттестации, и типы выходных результатов данных задач, которые должны быть документально оформлены.

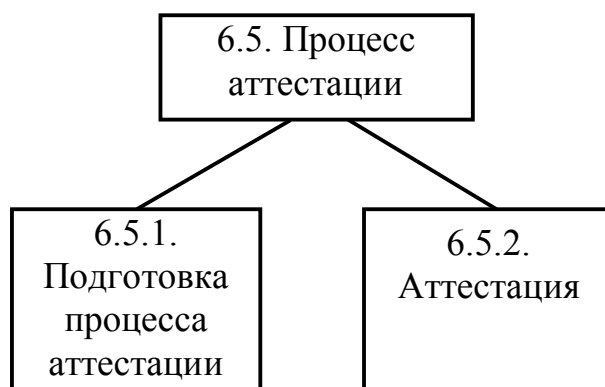


Рис. 18. Структура процесса аттестации

Таблица 22

Задачи работ процесса аттестации

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
<b>6.5.1. Задачи работы <i>Подготовка процесса аттестации</i></b>		
6.5.1.1	Определение необходимости в проекте работ по аттестации и степени их организационной независимости	
6.5.1.2	Установка процесса, задач, методов, методик и средств аттестации (при необходимости)	Описание (в составе результата задачи 6.5.1.4)
6.5.1.3	Выбор соответствующей независимой квалифицированной организации (при необходимости)	
6.5.1.4	Разработка плана аттестации	План
6.5.1.5	Реализация плана проведения аттестации, обеспечение доступности его результатов заказчику, при необходимости организация связи с процессом решения проблем (см. п. 2.3.8 пособия)	Отчет
<b>6.5.2. Задачи работы <i>Аттестация</i></b>		
6.5.2.1	Подготовка выбранных требований к испытаниям (тестированию), контрольных примеров и технических условий испытаний к анализу результатов испытаний	Описание
6.5.2.2	Обеспечение соответствия требований к испытаниям (тестированию), контрольных примеров и технических условий испытаний требованиям к объектам аттестации	

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
6.5.2.3	Проведение испытаний	Протокол
6.5.2.4	Подтверждение соответствия программного продукта заданным возможностям	Протокол
6.5.2.5	Проведение испытаний программного продукта в заданном контексте использования	Протокол

### 2.3.6. Процесс совместного анализа (The Joint Review Process)

*Процесс совместного анализа* является процессом оценки состояний и результатов работ по проекту. Совместные анализы проводятся в течение всего договора и применяются как на уровне управления проектом, так и на уровне его технической реализации. Данный процесс может выполняться двумя любыми сторонами, участвующими в договоре, когда одна сторона (анализирующая) проверяет другую (анализируемую).

Процесс совместного анализа состоит из *трех работ* (рис. 19). Общее число задач по данным работам равно 8.

Табл. 23 содержит задачи, реализуемые при выполнении соответствующих работ процесса совместного анализа, и типы выходных результатов данных задач, которые должны быть документально оформлены.



Рис. 19. Структура процесса совместного анализа

## Задачи работ процесса совместного анализа

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
<b>6.6.1. Задачи работы <i>Подготовка процесса совместного анализа</i></b>		
6.6.1.1	Организация проведения периодических и целевых анализов хода работ в установленные сроки	
6.6.1.2	Согласование ресурсов, необходимых для проведения анализа, между участвующими в нем сторонами	Протокол
6.6.1.3	Согласование плана, объема, процедур и критериев проведения анализа, состава анализируемых результатов работ	Протокол
6.6.1.4	Передача выявленных при проведении анализа проблем в процесс решения проблем (см. п. 2.3.8 пособия)	
6.6.1.5	Передача результатов анализа заинтересованным сторонам	
6.6.1.6	Согласование результатов анализа, принимаемых обязательств и критериев завершения анализа	Протокол
<b>6.6.2. Задачи работы <i>Анализы управления проектом</i></b>		
6.6.2.1	Оценка соответствия состояния проекта проектным планам, графикам, стандартам и руководствам	Отчет, протокол
<b>6.6.3. Задачи работы <i>Технические анализы</i></b>		
6.6.3.1	Оценка соответствия состояния создаваемых программных продуктов или услуг проектным планам, графикам, стандартам и руководствам	Отчет

### 2.3.7. Процесс аудита (The Audit Process)

*Процесс аудита* является процессом определения соответствия требованиям, планам и условиям договора. Данный процесс может выполняться двумя сторонами, участвующими в договоре, когда одна сторона (ревизирующая) проверяет другую сторону (ревизируемую).

Процесс аудита состоит из *двух работ* (рис. 20). Общее число задач по данным работам равно 8.

Табл. 24 содержит задачи, реализуемые при выполнении соответствующих работ процесса аудита, и типы выходных результатов данных задач, которые должны быть документально оформлены.

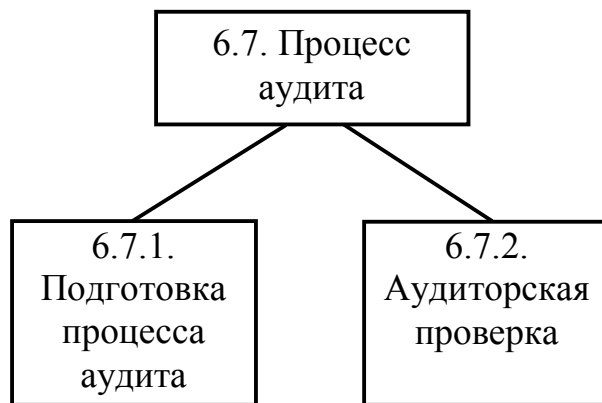


Рис. 20. Структура процесса аудита

Таблица 24

Задачи работ процесса аудита

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
<b>6.7.1. Задачи работы <i>Подготовка процесса аудита</i></b>		
6.7.1.1	Организация проведения аудиторских проверок в сроки согласно плану	
6.7.1.2	Обеспечение независимости аудиторского персонала	
6.7.1.3	Согласование ресурсов, необходимых для проведения аудита, между участвующими в нем сторонами	Протокол
6.7.1.4	Согласование плана, объема, процедур и критериев проведения аудиторской проверки, состава проверяемых результатов работ	Протокол
6.7.1.5	Передача выявленных при проведении аудиторской проверки проблем в процесс решения проблем (см. п. 2.3.8 пособия)	Протокол
6.7.1.6	Передача результатов аудиторской проверки заинтересованным сторонам	
6.7.1.7	Согласование результатов, принимаемых обязательств и критериев завершения аудиторской проверки	Протокол
<b>6.7.2. Задачи работы <i>Аудиторская проверка</i></b>		
6.7.2.1	Проведение аудиторской проверки по критериям	Отчет

*Критерии проведения аудиторской проверки* (см. задачу 6.7.2.1):

- соответствие запрограммированных программных продуктов (объектов) проектной документации;
- пригодность подготовки приемки и требований к тестированию, установленных в документации, для приемки программных продуктов (объектов);
- соответствие тестовых данных установленным техническим требованиям;
- успешность тестирования программных продуктов и их соответствие установленным к ним требованиям;
- правильность отчетов об испытаниях (тестировании), устранение расхождений между фактическими и ожидаемыми результатами;
- соответствие документации пользователя установленным стандартам;
- выполнение работ в соответствии с утвержденными требованиями, планами и договором;
- соответствие стоимости и графика проведения работ утвержденным планам.

### **2.3.8. Процесс решения проблем (The Problem Resolution Process)**

*Процесс решения проблем* является процессом анализа и решения проблем (включая обнаруженные несоответствия), которые обнаружены в ходе выполнения разработки, эксплуатации, сопровождения или других процессов.

Данный процесс состоит из *двух работ* (рис. 21). Общее число задач по данным работам равно 2.



Рис. 21. Структура процесса решения проблем

Табл. 25 содержит задачи, реализуемые при выполнении соответствующих работ процесса решения проблем, и типы выходных результатов данных задач, которые должны быть документально оформлены.

## Задачи работ процесса решения проблем

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
6.8.1. Задачи работы <i>Подготовка процесса решения проблем</i>		
6.8.1.1	Установка процесса решения проблем для обработки всех проблем, выявленных в программных продуктах и работах	Протокол
6.8.2. Задачи работы <i>Решение проблемы</i>		
6.8.2.1	Отслеживание проблем путем их выявления, исследования, анализа и решения и их документирование	Отчет

## 2.4. Организационные процессы жизненного цикла

Организационные процессы жизненного цикла программных средств и систем представлены на рис. 22. На данном рисунке сохранена нумерация процессов, принятая в ИСО/МЭК 12207.

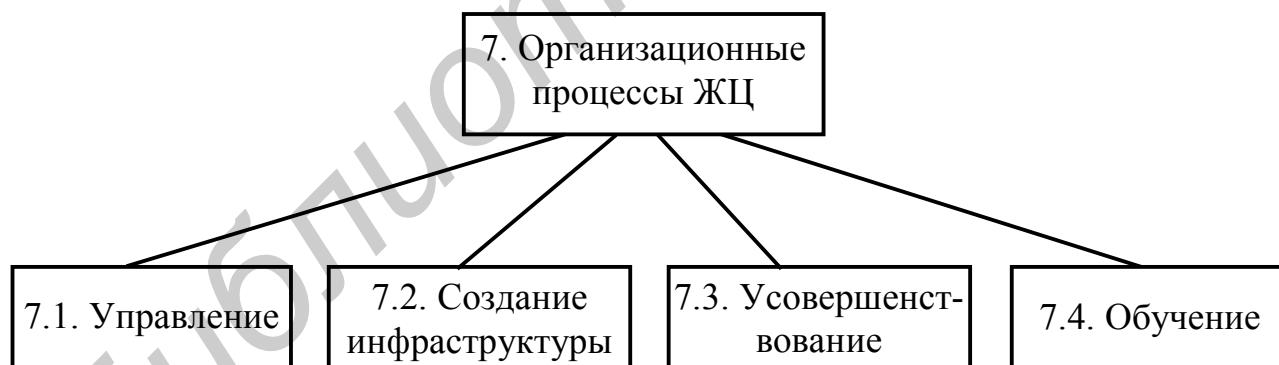


Рис. 22. Организационные процессы жизненного цикла программных средств и систем

### 2.4.1. Процесс управления (The Management Process)

*Процесс управления* состоит из общих работ и задач, которые могут быть использованы любой стороной, управляющей соответствующим процессом. За



управление продуктом, проектом, работами и задачами основных и вспомогательных процессов отвечает *администратор*.

Процесс управления состоит из *пяти работ* (рис. 23). Общее число задач по данным работам равно 12.

Табл. 26 содержит задачи, реализуемые при выполнении соответствующих работ процесса управления, и типы выходных результатов данных задач, которые должны быть документально оформлены.

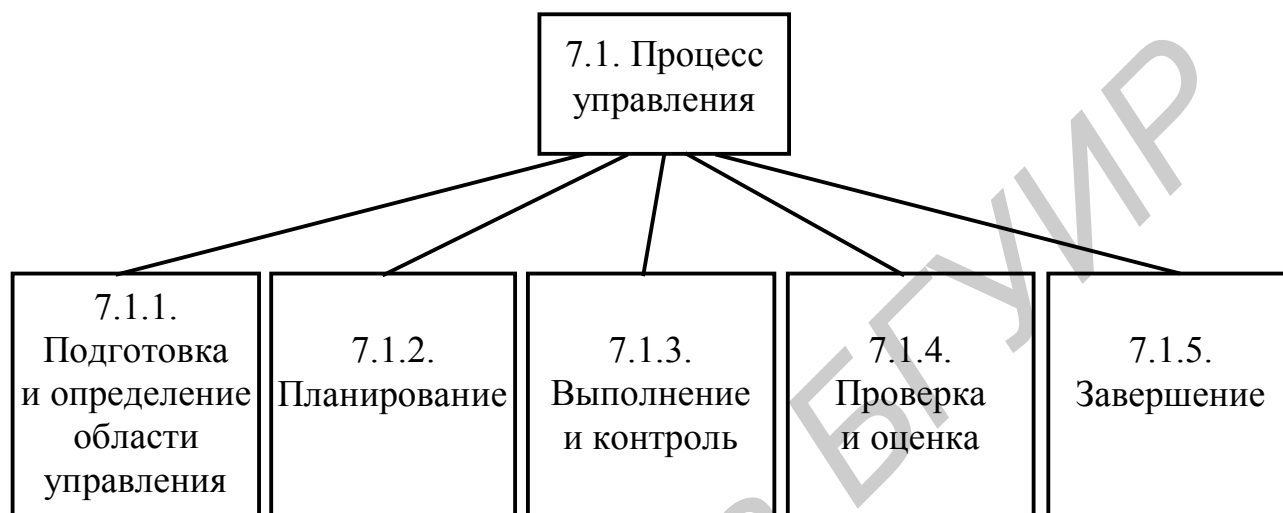


Рис. 23. Структура процесса управления

Таблица 26

Задачи работ процесса управления

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
<b>7.1.1. Задачи работы <i>Подготовка и определение области управления</i></b>		
7.1.1.1	Установка требований к реализуемому процессу	Описание
7.1.1.2	Определение возможности реализации процесса, проверка соответствия ресурсов и сроков	Описание
7.1.1.3	Изменение требований к процессу для удовлетворения критериев его завершения (при необходимости)	Протокол
<b>7.1.2. Задачи работы <i>Планирование</i></b>		
7.1.2.1	Разработка планов выполнения процессов	План
<b>7.1.3. Задачи работы <i>Выполнение и контроль</i></b>		
7.1.3.1	Управление процессом на базе разработанного плана	

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
7.1.3.2	Текущий надзор за выполнением процесса	Отчет (в составе результата задачи 7.1.3.4)
7.1.3.3	Исследование и решение проблем, обнаруженных при выполнении процесса, при необходимости изменение планов	Отчет (в составе результата 7.1.3.4)
7.1.3.4	Отчет в установленные сроки о реализации процесса	Отчет
<b>7.1.4. Задачи работы <i>Проверка и оценка</i></b>		
7.1.4.1	Обеспечение оценки программных продуктов и планов на соответствие требованиям	
7.1.4.2	Проверка результатов оценок программных продуктов, работ и задач на соответствие целям и планам	Отчет
<b>7.1.5. Задачи работы <i>Завершение</i></b>		
7.1.5.1	Оценка всех программных продуктов, работ и задач по критериям, установленным в договоре	Отчет
7.1.5.2	Контроль результатов и полноты документации созданных программных продуктов и выполненных работ, передача их в архив	

**Планы выполнения процессов** разрабатываются администратором проекта в ходе решения задачи 7.1.2.1. Они должны охватывать следующие вопросы:

- установка графиков выполнения задач;
- оценка необходимых трудозатрат;
- определение ресурсов, необходимых для выполнения задач;
- распределение задач по исполнителям;
- определение обязанностей исполнителей;
- определение критических ситуаций, связанных с задачами или самим процессом;
- установка используемых в процессе критериев управления качеством;
- определение затрат, связанных с реализацией процесса;
- обеспечение условий и определение инфраструктуры выполнения процесса.

## 2.4.2. Процесс создания инфраструктуры (The Infrastructure Process)

*Процесс создания инфраструктуры* является процессом установления и сопровождения инфраструктуры, необходимой для любого другого процесса. *Инфраструктура* содержит технические и программные средства, инструментальные средства, методики, стандарты и условия для разработки, эксплуатации или сопровождения.

Процесс создания инфраструктуры состоит из *трех работ* (рис. 24). Общее число задач по данным работам равно 5.



Рис. 24. Структура процесса создания инфраструктуры

Табл. 27 содержит задачи, реализуемые при выполнении соответствующих работ процесса создания инфраструктуры, и типы выходных результатов данных задач, которые должны быть документально оформлены.

Таблица 27

Задачи работ процесса создания инфраструктуры

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
<b>7.2.1. Задачи работы <i>Подготовка процесса создания инфраструктуры</i></b>		
7.2.1.1	Определение инфраструктуры, удовлетворяющей требованиям к процессу	Описание (в составе результата задачи 7.2.1.2)
7.2.1.2	Разработка плана создания установленной инфраструктуры	План

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
7.2.2. Задачи работы <i>Создание инфраструктуры</i>		
7.2.2.1	Разработка конфигурации инфраструктуры с учетом характеристик продукта и процесса	Описание
7.2.2.2	Создание инфраструктуры к требуемому сроку	
7.2.3. Задачи работы <i>Сопровождение инфраструктуры</i>		
7.2.3.1	Сопровождение, контроль и изменение инфраструктуры для удовлетворения требований к процессу, обеспечение связи с процессом управления конфигурацией (см. п. 2.3.2 пособия)	Описание

### 2.4.3. Процесс усовершенствования (The Improvement Process)

*Процесс усовершенствования* является процессом установления, оценки, измерения, контроля и улучшения любого процесса жизненного цикла программных средств.

Процесс усовершенствования состоит из *трех работ* (рис. 25). Общее число задач по данным работам равно 6.

Табл. 28 содержит задачи, реализуемые при выполнении соответствующих работ процесса усовершенствования, и типы выходных результатов данных задач, которые должны быть документально оформлены.

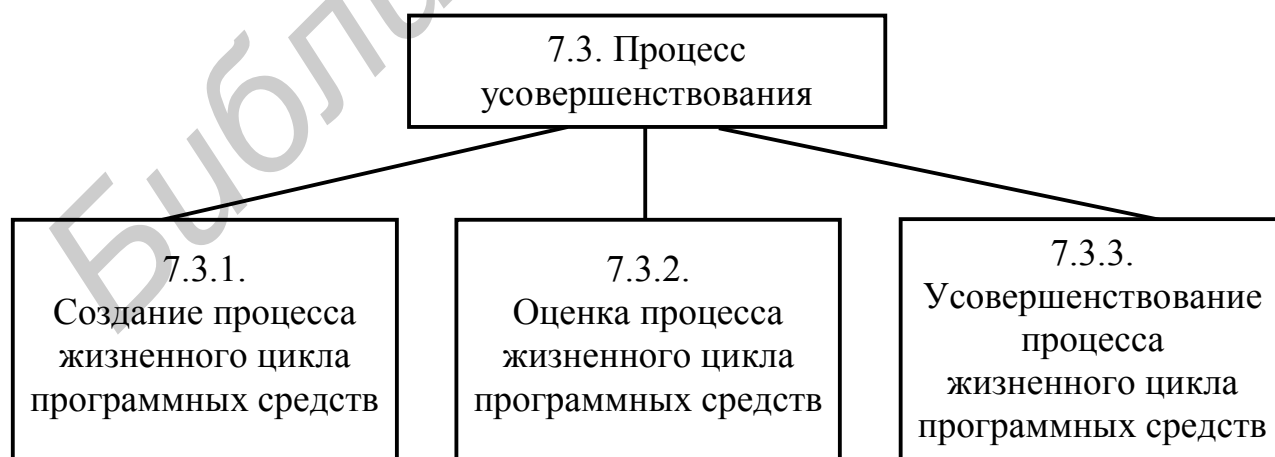


Рис. 25. Структура процесса усовершенствования

## Задачи работ процесса усовершенствования

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
7.3.1. Задачи работы <i>Создание процесса</i>		
7.3.1.1	Определение набора организационных процессов для всех процессов ЖЦ ПС и механизма управления процессом усовершенствования	Процедура
7.3.2. Задачи работы <i>Оценка процесса</i>		
7.3.2.1	Разработка и применение процедуры оценки процесса	Процедура
7.3.2.2	Планирование и выполнение анализов процессов в установленные сроки	План, отчет
7.3.3. Задачи работы <i>Усовершенствование процесса</i>		
7.3.3.1	Усовершенствование процесса (при необходимости)	Отчет
7.3.3.2	Сбор и анализ архивных, технических и оценочных данных по выполняемым процессам для их усовершенствования	
7.3.3.3	Сбор, обновление и использование данных о расходах для усовершенствования организационных процессов административной деятельности	

#### 2.4.4. Процесс обучения (The Training Process)

*Процесс обучения* является процессом обеспечения первоначального и продолженного обучения персонала работам по заказу, поставке, разработке, эксплуатации или сопровождению программного проекта.

Процесс обучения состоит из *трех работ* (рис. 26). Общее число задач по данным работам равно 4.

Табл. 29 содержит задачи, реализуемые при выполнении соответствующих работ процесса обучения, и типы выходных результатов данных задач, которые должны быть документально оформлены.



Рис. 26. Структура процесса обучения

Таблица 29

Задачи работ процесса обучения

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
<b>7.4.1. Задачи работы <i>Подготовка процесса обучения</i></b>		
7.4.1.1	Анализ требований к проекту для определения необходимости обучения персонала, разработка плана обучения	План
<b>7.4.2. Задачи работы <i>Разработка учебных материалов</i></b>		
7.4.2.1	Разработка руководств и материалов для обучения	Описание
<b>7.4.3. Задачи работы <i>Реализация плана обучения</i></b>		
7.4.3.1	Реализация плана обучения с сохранением протоколов	Протокол
7.4.3.2	Обеспечение своевременной подготовки персонала к выполнению запланированных работ и задач	

## 2.5. Адаптация требований стандарта ИСО/МЭК 12207 к условиям проекта

Стандарт *ИСО/МЭК 12207* определяет общий случай разработки типового проекта. В данном стандарте регламентируется *общее число процессов* жизненного цикла программных средств, равное 17, *общее число работ*, равное 74, *общее число задач*, равное 232.

Однако при разработке конкретных проектов может отсутствовать необходимость в использовании тех или иных процессов, работ или задач ЖЦ ПС.

Кроме того, может потребоваться ввести в ЖЦ дополнительные процессы, работы и задачи, не регламентированные стандартом *ИСО/МЭК 12207*. С учетом этого в интересах сокращения стоимости и улучшения качества конкретного проекта требования данного стандарта следует адаптировать. Процесс адаптации определен в *Приложении А ИСО/МЭК 12207*.

**Процесс адаптации** является процессом применения положений данного стандарта к условиям реализации конкретного программного проекта. Данный процесс следует относить к *дополнительным процессам жизненного цикла*.

Процесс адаптации состоит из *четырёх работ* (рис. 27). Номера данных работ соответствуют принятым в стандарте *ИСО/МЭК 12207*. Общее число задач по работам равно 6.

Табл. 30 содержит задачи, реализуемые при выполнении соответствующих работ процесса адаптации, и типы выходных результатов данных задач, которые должны быть документально оформлены.

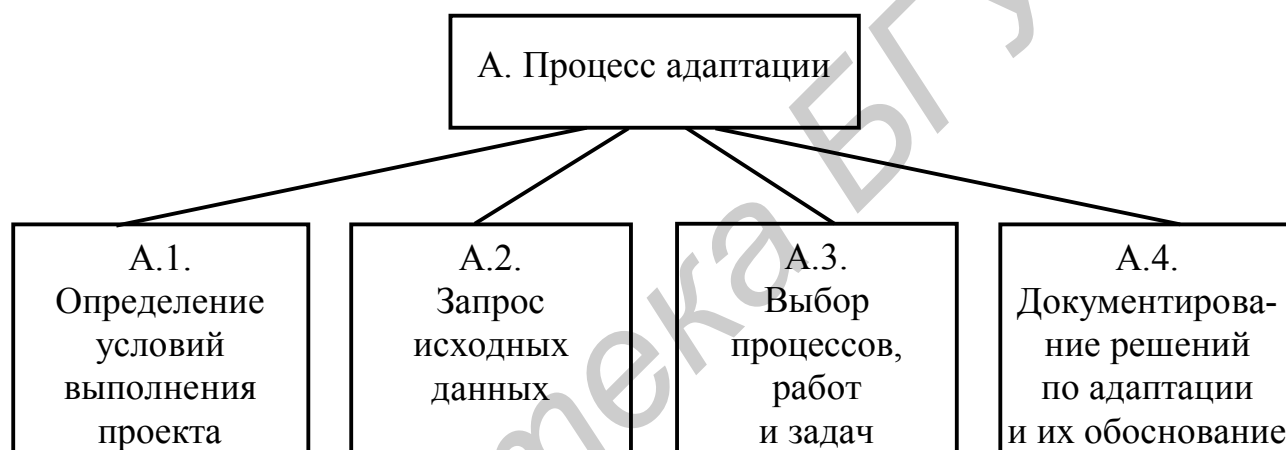


Рис. 27. Структура процесса адаптации

Таблица 30

Задачи работ процесса адаптации

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
А.1. Задачи работы <b>Определение условий выполнения проекта</b>		
А.1.1	Определение характеристик условий выполнения проекта, влияющих на адаптацию	
А.2. Задачи работы <b>Запрос исходных данных</b>		
А.2.1	Запрос и получение исходных данных, влияющих на адаптацию, от участвующих в проекте организаций	

№ задачи в ИСО/МЭК 12207	Описание задачи	Тип выходного результата задачи
<b>А.3. Задачи работы <i>Выбор процессов, работ и задач</i></b>		
А.3.1	Определение необходимых процессов, работ и задач с учетом разрабатываемой документации и обязанностей исполнителей	
А.3.2	Установка в договоре дополнительных процессов, работ и задач, не описанных в стандарте ИСО/МЭК 12207, оценка соответствия организационных процессов	
А.3.3	Отдельный анализ на предмет исключения из ЖЦ обязательных (по стандарту ИСО/МЭК 12207) задач, с учетом факторов риска, стоимости, графика работ, выполнимости, объема, критичности и интерфейса с пользователем	
<b>А.4. Задачи работы <i>Документирование решений по адаптации и их обоснование</i></b>		
А.4.1	Документальное оформление решений по адаптации с их обоснованием	Отчет

К характеристикам условий выполнения проекта (см. задачу А.1.1) могут быть отнесены следующие:

- модель жизненного цикла;
- влияние жизненного цикла существующей системы;
- требования к системе и программным средствам;
- организационные подходы, процедуры и цели;
- размер, сложность, критичность и типы системы, программного продукта или программной услуги;
- методы разработки;
- количество задействованного персонала и участвующих в проекте сторон;
- методы и политика заказа.

Приложение В стандарта ИСО/МЭК 12207 содержит руководство по адаптации, поясняющее использование процесса адаптации к конкретному программному проекту.

В данном приложении рекомендуется выполнять два уровня адаптации ИСО/МЭК 12207:

*первый уровень* – адаптация к конкретной области деятельности – медицинской, авиационной, военной и т.п.;

*второй уровень* – адаптация к каждому конкретному проекту или договору.



*Приложение В* может быть использовано для первого уровня адаптации. При этом рекомендуется:

- для встроенного в систему программного продукта определить, требуется ли от разработчика выполнение работ по созданию системы;
- для отдельно поставляемого продукта рассмотреть работы по созданию системы (см. работы 5.3.2, 5.3.3, 5.3.10, 5.3.11 на рис. 7), хотя они могут и не понадобиться.

*Приложение В* рекомендует также выполнять адаптацию оценок программных продуктов и работ, проводимых в течение жизненного цикла. Данные оценки группируются в **пять категорий**:

- 1) *оценки внутри основных процессов*; выполняются персоналом, реализующим данные процессы; цель – текущая оценка полученных результатов;
- 2) *верификация и аттестация* (см. рис. 17, 18); выполняется заказчиком, поставщиком или независимой стороной; цель – верификация и аттестация с различной степенью зависимости от проекта;
- 3) *совместные анализы и аудиторские проверки* (см. рис. 19, 20); выполняются проверяющей и проверяемой сторонами; цель – оценка состояния и ответственности продуктов и работ утвержденному графику;
- 4) *обеспечение качества* (см. рис. 16); выполняется персоналом, не зависящим от разработчиков программного продукта; цель – представление независимой гарантии соответствия программных продуктов и процессов требованиям договора и утвержденным планам;
- 5) *усовершенствование* (см. рис. 25); выполняется организацией без учета требований конкретного проекта или договора; цель – эффективное управление реализуемыми процессами и их усовершенствование.

Четыре первых категории оценок выполняются на проектном уровне, пятая категория – на организационном уровне. Данные категории оценок следует выбирать и адаптировать в соответствии с областью действия, величиной, сложностью и критичностью проекта, а также с учетом подходов к управлению процессами, существующих в организации.

Очевидно, что вопросы адаптации и применения *ИСО/МЭК 12207* тесно связаны с **основными характеристиками проекта**. К ним относятся:

- 1) *организационные подходы*; например, к машинным языкам, безопасности, защите, требованиям по резервированию технических средств и управлению риском;
- 2) *политика заказа*; например, типы договора, наличие подрядчиков, привлечение субподрядчиков и посредников по верификации и аттестации;
- 3) *концепция поддержки*; например, ожидаемая длительность поддержки, степень изменения продукта, сторона, выполняющая поддержку;
- 4) *модель жизненного цикла*; например, каскадная, эволюционная, инкрементная;
- 5) *вовлеченные стороны*; например, заказчик, поставщик, разработчик, субподрядчик, посредники по верификации и аттестации, персонал сопровождения; численность сторон; для больших проектов (десятки и сотни лиц) требу-

ется соответствующий административный надзор и контроль, оценки, анализы, аудиторские проверки; для малых проектов такой контроль может не понадобиться;

6) *работы жизненного цикла системы*; например, подготовка проекта заказчика, разработка и сопровождение поставщиком;

7) *характеристики системного уровня*; например, количество подсистем и объектов конфигурации;

8) *характеристики программного уровня*; например, количество программных объектов, типы, объемы и критичность программных продуктов, технические риски; выделяются следующие типы программных продуктов:

- новая разработка; должны учитываться все требования к процессу разработки;
- использование готового программного продукта; должна быть выполнена оценка функциональных характеристик, документации, применимости, возможность поддержки; процесс разработки может не понадобиться;
- модификация готового программного продукта; должна быть выполнена оценка функциональных характеристик, документации, применимости, возможность поддержки; процесс разработки реализуется с учетом критичности продукта и величины изменений;
- программный или программно-аппаратный продукт, встроенный или подключенный к системе; необходимо учитывать работы процесса разработки, связанные с системой;
- отдельно поставляемый программный продукт; не требуется учитывать работы процесса разработки, связанные с системой;
- непоставляемый программный продукт; требования стандарта ИСО/МЭК 12207 можно не учитывать;

9) *другие характеристики*; например, усиленный административный контроль за критичными или большими программными продуктами.

## **2.6. Инструментальные средства автоматизации жизненного цикла программных средств и систем**

Очевидно, что большие размеры и высокая сложность разрабатываемых программных средств при ограничениях на бюджетные и временные затраты проекта могут привести к низкому качеству конечных программных продуктов и системы в целом. В этой связи в последнее время все большее внимание уделяется современным технологиям и инструментальным средствам, обеспечивающим автоматизацию процессов жизненного цикла программных средств (CASE-средствам). Использование таких инструментальных средств позволяет существенно сократить длительность и стоимость разработки систем и ПС при одновременном повышении качества процесса разработки и, как следствие, ка-

чества разработанных программных средств.

К современным инструментальным средствам, обеспечивающим эффективную поддержку процессов жизненного цикла программных средств и систем, можно отнести семейство интегрированных продуктов компании *Telelogic*. Основными в данном семействе являются следующие инструментальные средства.

### 2.6.1. Telelogic DOORS

Инструментальное средство *Telelogic DOORS* предназначено для автоматизации управления требованиями в жизненном цикле программных средств.

*Telelogic DOORS* может быть использовано при выполнении процессов заказа, поставки и разработки. Табл. 31 содержит работы и задачи вышеназванных процессов, которые может поддерживать *Telelogic DOORS*.

В данной таблице номера работ и задач ЖЦ ПС приведены в соответствии с их иерархическими номерами в стандарте *ISO/IEC 12207:1995* и его аутентичных аналогах *СТБ ИСО/МЭК 12207–2003* и *ГОСТ Р ИСО/МЭК 12207–99* [17, 12, 5]. В последнем столбце таблицы приведены номера таблиц пособия, содержащих описание соответствующих работ процессов жизненного цикла программных средств.

Таблица 31

Работы и задачи процессов заказа, поставки и разработки,  
поддерживаемые инструментальным средством *Telelogic DOORS*

Название процесса	Номер и название работы	Номер и содержание поддерживаемой задачи	Номер таблицы пособия
Заказ	5.1.1. Подготовка процесса заказа	5.1.1.1. Описание концепции или потребности в заказе 5.1.1.2. Анализ требований к системе 5.1.1.4. Анализ требований к программным средствам	1
Поставка	5.2.4. Планирование	5.2.4.1. Анализ требований к заказу	2
Разработка	5.3.2. Анализ требований к системе	5.3.2.1. Анализ области применения системы и определение требований к ней	4
	5.3.3. Проектирование системной архитектуры	5.3.3.1. Распределение требований к системе между объектами технических и программных средств архитектуры и ручными операциями и дальнейшее уточнение требований	5

Название процесса	Номер и название работы	Номер и содержание поддерживаемой задачи	Номер таблицы пособия
	5.3.4. Анализ требований к программным средствам	5.3.4.1. Определение требований к программным средствам	6
	5.3.5. Проектирование программной архитектуры	5.3.5.1. Распределение требований к программному объекту между его компонентами и дальнейшее уточнение требований	7
	5.3.6. Техническое проектирование программных средств	5.3.6.1. Распределение технических требований к компонентам между программными модулями и дальнейшее уточнение требований	8

Вышеприведенные задачи (табл. 31) непосредственно связаны с разработкой требований к системе, программным средствам или их элементам и могут быть автоматизированы с помощью *Telelogic DOORS*. Это позволяет повысить качество выполнения данных задач. Кроме того, следует иметь в виду, что для большинства других задач процессов жизненного цикла требования являются исходными данными. Поэтому структуризация требований и автоматизация их разработки также повышает качество выполнения таких задач, а следовательно, и всего процесса разработки систем и программных средств в целом.

## 2.6.2. Telelogic TAU

Семейство инструментальных средств *Telelogic TAU* предназначено для автоматизации работ и задач процесса разработки программных средств и систем. Данное семейство предоставляет визуальную среду разработки, основанную на использовании универсального языка моделирования UML, с возможностью последующей автоматической кодогенерации ПС. Использование семейства *Telelogic TAU* позволяет унифицировать язык общения между системными аналитиками, проектировщиками, программистами и другими разработчиками, ускорить и повысить качество проектирования ПС и систем, автоматизировать тестирование (испытания) программных модулей, компонентов, ПС и систем.

Семейство *Telelogic TAU* включает следующие инструментальные средства:

- *TAU/Model Author* – инструментальное средство для рисования диаграмм моделей;
- *TAU/Architect* – инструментальное средство для системных инженеров;

- **TAU/Developer** – инструментальное средство для разработчиков программных средств;

- **TAU/Tester** – инструментальное средство для тестировщиков.

Табл. 32 содержит работы и задачи процесса разработки, которые поддерживает семейство *Telelogic TAU*.

Таблица 32

Работы и задачи процесса разработки,  
поддерживаемые семейством инструментальных средств *Telelogic TAU*

Номер и название работы	Номер и содержание поддерживаемой задачи	Номер таблицы пособия
5.3.3. Проектирование системной архитектуры	5.3.3.1. Определение общей архитектуры системы	5
5.3.5. Проектирование программной архитектуры	5.3.5.1. Преобразование требований к программному объекту в его архитектуру (разработка эскизного проекта) 5.3.5.2. Разработка эскизного проекта интерфейсов программного объекта и его компонентов 5.3.5.3. Разработка эскизного проекта базы данных	7
5.3.6. Техническое проектирование программных средств	5.3.6.1. Разработка технического проекта для компонентов программного объекта (представление их в виде набора программных модулей) 5.3.6.2. Разработка технического проекта интерфейсов программного объекта, его компонентов и модулей 5.3.6.3. Разработка технического проекта базы данных	8
5.3.7. Программирование и тестирование программных средств	5.3.7.1. Программирование программных модулей, базы данных 5.3.7.2. Тестирование программных модулей и базы данных	9
5.3.8. Сборка программных средств	5.3.8.2. Сборка программных модулей и компонентов и тестирование результатов сборки	10
5.3.9. Квалификационные испытания программных средств	5.3.9.1. Проведение квалификационных испытаний (тестирования) программных средств	11
5.3.11. Квалификационные испытания системы	5.3.11.1. Проведение квалификационных испытаний	13

### 2.6.3. Telelogic SYNERGY

Инструментальное средство *Telelogic SYNERGY* предназначено для автоматизации управления изменениями и конфигурацией программных средств. Его использование позволяет упростить организацию коллективной работы над проектом и обеспечить контроль текущего состояния проекта с учетом возможных изменений в требованиях заказчика, в функциях разрабатываемых программных продуктов, исправлений ошибок в продуктах.

*Telelogic SYNERGY* обеспечивает поддержку изменений, вносимых в выходные результаты работ процессов разработки или сопровождения, автоматизирует процесс управления конфигурацией жизненного цикла ПС и систем (см. п. 2.3.2 пособия). *Telelogic SYNERGY* также может быть использован при выполнении ряда задач других процессов ЖЦ (табл. 33).

Таблица 33

Работы и задачи процессов жизненного цикла программных средств, поддерживаемые инструментальным средством *Telelogic SYNERGY*

Название процесса	Номер и название работы	Номер и содержание поддерживаемой задачи	Номер таблицы пособия
Заказ	5.1.3. Подготовка и корректировка договора	5.1.3.5. Контроль изменений, вносимых в договор в ходе его реализации	1
	5.1.5. Приемка и закрытие договора	5.1.5.3. Управление конфигурацией поставленного ПП	
Поставка	5.2.5. Выполнение и контроль	5.2.5.3. Надзор за реализацией проекта, выявление и решение проблем	2
Документирование	6.1.3. Выпуск	6.1.3.2. Управление документированием в соответствии с процессом управления конфигурацией	18
	6.1.4. Сопровождение	6.1.4.1. Внесение изменений в документацию согласно процессам сопровождения и управления конфигурацией	18
Создание инфраструктуры	7.2.3. Сопровождение инфраструктуры	7.2.3.1. Сопровождение, контроль и изменение инфраструктуры для удовлетворения требований к процессу, обеспечение связи с процессом управления конфигурацией	27

## 2.6.4. Telelogic DocExpress

Инструментальное средство *Telelogic DocExpress* предназначено для поддержки процесса документирования жизненного цикла программных средств (см. п. 2.3.1 пособия). *Telelogic DocExpress* позволяет автоматизировать поступление из различных источников исходных данных для документирования, поддерживать исходные данные и документацию в актуальном состоянии, выполнять автоматическое форматирование документов в соответствии с действующими стандартами на документацию, упрощает сопровождение документов.

*Telelogic DocExpress* может использоваться при выполнении задач жизненного цикла, в которых предполагается документальное оформление выходных результатов (табл. 1 – 30 пособия, правый столбец).

## 2.6.5. Telelogic TAU Logiscope

Семейство инструментальных средств *Telelogic TAU Logiscope* входит в состав семейства *Telelogic TAU* и может использоваться инженерами по качеству и тестировщиками.

Семейство *TAU Logiscope* предназначено для поддержки задач жизненного цикла, связанных с выполнением проверок, оценок и тестирования промежуточных продуктов разработки. Применение *Telelogic TAU Logiscope* позволяет управлять качеством процессов, промежуточных и конечных продуктов жизненного цикла программных средств, сократить стоимость и сроки выполнения проектов.

*Telelogic TAU Logiscope* дает возможность автоматизировать поиск ошибок в исходных кодах программных модулей, компонент, объектов программных средств, оценить качество, сложность и соответствие принятым правилам исходных кодов, оценить степень покрытия исходных кодов тестами.

Семейство *Telelogic TAU Logiscope* включает следующие инструментальные средства:

- ***Logiscope Audit*** – инструментальное средство для оценки качества и графического анализа исходных программных кодов;
- ***Logiscope RuleChecker*** – инструментальное средство для проверки исходного кода на соответствие принятым правилам;
- ***Logiscope TestChecker*** – инструментальное средство для проверки степени покрытия исходного кода тестовыми наборами;
- ***Logiscope Reviewer*** – инструментальное средство, сочетающее в себе возможности *Logiscope Audit* и *Logiscope RuleChecker*.

Семейство *Telelogic TAU Logiscope* поддерживает вспомогательные процессы жизненного цикла, связанные с управлением качеством процессов и продуктов: обеспечения качества, верификации, аттестации, совместного анализа, аудита (см. рис. 3 и пп. 2.3.3 – 2.3.7 пособия).

Табл. 34 содержит работы и задачи процессов поставки и разработки, при

выполнении которых может использоваться семейство *Telelogic TAU Logiscope*. Следует обратить внимание, что данное семейство используется при выполнении работ и задач процесса разработки, связанных с исходными текстами программных модулей. Это касается, в первую очередь, работы «Программирование и тестирование программных средств» и, возможно, начальных этапов работы «Сборка программных средств», когда программные модули собираются в небольшие группы и выполняется тестирование этих групп.

Таблица 34

Работы и задачи процессов поставки и разработки,  
поддерживаемые семейством инструментальных средств  
*Telelogic TAU Logiscope*

Название процесса	Номер и название работы	Номер и содержание поддерживаемой задачи	Номер таблицы пособия
Поставка	5.2.6. Проверка и оценка	5.2.6.1. Участие в совещаниях, подготовке приемки ПП, приемочных испытаниях, совместных анализах и аудиторских проверках 5.2.6.2. Выполнение верификации и аттестации ПП, программной услуги и процессов 5.2.6.3. Выполнение работ по обеспечению качества	2
Разработка	5.3.7. Программирование и тестирование программных средств	5.3.7.1. Программирование программных модулей, базы данных, разработка процедур и данных для их испытаний (тестирования) 5.3.7.4. Уточнение общих требований к тестированию и программы сборки программного средства 5.3.7.5. Оценка результатов программирования и тестирования по критериям	9
	5.3.8. Сборка программных средств	5.3.8.5. Оценка планов и результатов сборки и тестирования по критериям	10

В заключение следует отметить, что инструментальные средства семейства компании *Telelogic* могут интегрироваться друг с другом. Таким образом, их совместное использование поддерживает подавляющее большинство работ и задач процессов жизненного цикла программных средств.



## 3. ДОПОЛНЕНИЯ ISO/IEC 12207:1995 / AMD.1 И ISO/IEC 12207:1995 / AMD.2

### 3.1. Общие сведения

В 2002 г. введено в действие Дополнение к базовому стандарту *ISO/IEC 12207:1995* [17] под обозначением *ISO/IEC 12207:1995 – Информационная технология – Процессы жизненного цикла программных средств / Amd.1: 2002* [18]. Данное Дополнение расширяет базовый стандарт и обеспечивает согласование его положений с положениями других стандартов, связанных с жизненным циклом программных средств. В первую очередь это касается стандартов, относящихся к оценке процессов и продуктов ЖЦ ПС (*ISO/IEC 15504–1–8* [30 – 37], *ISO/IEC 14598–1–6* [21 – 26]), а также стандартов, посвященных технологиям, ориентированным на человека (например *ISO 13407:1999* [20]).

В 2004 г. введено в действие второе Дополнение к базовому стандарту *ISO/IEC 12207:1995* под обозначением *ISO/IEC 12207:1995 – Информационная технология – Процессы жизненного цикла программных средств / Amd.2:2004* [19]. В данном Дополнении отредактированы некоторые положения Дополнения *ISO/IEC 12207:1995/Amd.1:2002* и введены дальнейшие расширения базового стандарта.

Дополнениями *ISO/IEC 12207:1995/Amd.1:2002* и *ISO/IEC 12207:1995/Amd.2:2004* внесены следующие **изменения** в базовую структуру процессов и работ жизненного цикла программных средств, регламентированную стандартом *ISO/IEC 12207:1995*.

Во-первых, добавлено пять новых процессов ЖЦ ПС.

Во-вторых, некоторые существующие процессы и работы уточнены или расширены.

В-третьих, для отдельных процессов введено понятие подпроцессов. Подпроцессы по целевому признаку и выходным результатам объединяют соответствующие стороны существующих или новых работ.

В-четвертых, все процессы и подпроцессы ЖЦ ПС определены в терминах их целей и результатов.

Дополнение *Amd.1:2002* [18] содержит следующие **приложения**:

Приложение Е (информационное). Связь с *ISO/IEC 12207:1995*.

Приложение F (обязательное). Цель и результаты.

Приложение G (информационное). Структура процесса *ISO/IEC 12207:1995* для «новых» процессов из Приложения F.

Приложение H (информационное). *ISO/IEC TR 15504–2*, PDAM1, расширения эталонной модели на процесс заказа *ISO/IEC 12207:1995*.

В *Приложении Е* определена связь структуры процессов Дополнения *Amd.1:2002* с процессами *ISO/IEC 12207:1995*. С этой целью в Дополнении *Amd.1:2002* определены следующие **типы процессов (подпроцессов)**:

- базовый (basic, B) – эти процессы и подпроцессы соответствуют процессам и работам *ISO/IEC 12207:1995*;
- новый (new, N) – эти процессы и подпроцессы введены в дополнение к процессам, определенным в *ISO/IEC 12207:1995*;
- расширенный (extended, E) – эти процессы и подпроцессы являются уточнением или развитием существующих процессов или работ, определенных в *ISO/IEC 12207:1995*;
- составной (component, C) – эти подпроцессы представляют собой объединение существующих в *ISO/IEC 12207:1995* работ.

Структура процессов жизненного цикла программных средств, учитывающая положения Дополнений *Amd.1:2002* и *Amd.2:2004*, приведена на рис. 28. Здесь и на последующих рисунках данного раздела в скобках приведены типы соответствующих процессов, определенные в Дополнениях.

Как видно из сравнения рис. 28 и рис. 3, в жизненный цикл программных средств введено **пять новых процессов**:

- процесс практичности (удобства использования, применимости – The Usability Process);
- процесс человеческого ресурса (The Human Resource Process);
- процесс управления средствами (The Asset Management Process);
- процесс управления повторным использованием программ (The Reuse Program Management Process);
- процесс проектирования предметной области (The Domain Engineering Process).

Процесс практичности относится к группе вспомогательных процессов ЖЦ ПС, остальные четыре процесса – к группе организационных процессов.

Кроме новых процессов в структуру жизненного цикла программных средств введено два процесса, классифицированных Дополнениями *Amd.1:2002* и *Amd.2:2004* как «расширенные». К ним относятся:

- процесс управления заявками на изменения (The Change Request Management Process);
- процесс оценки продукта (The Product Evaluation Process).

К расширенным процессам относится также процесс управления решением проблем (The Problem Resolution Management Process), представляющий собой развитие процесса решения проблем из базового стандарта *ISO/IEC 12207:1995* (см. рис. 3 и 28).

Описание структуры новых процессов и подпроцессов жизненного цикла программных средств на уровне работ и задач приведено в *Приложении G ISO/IEC 12207:1995/Amd.1:2002*.

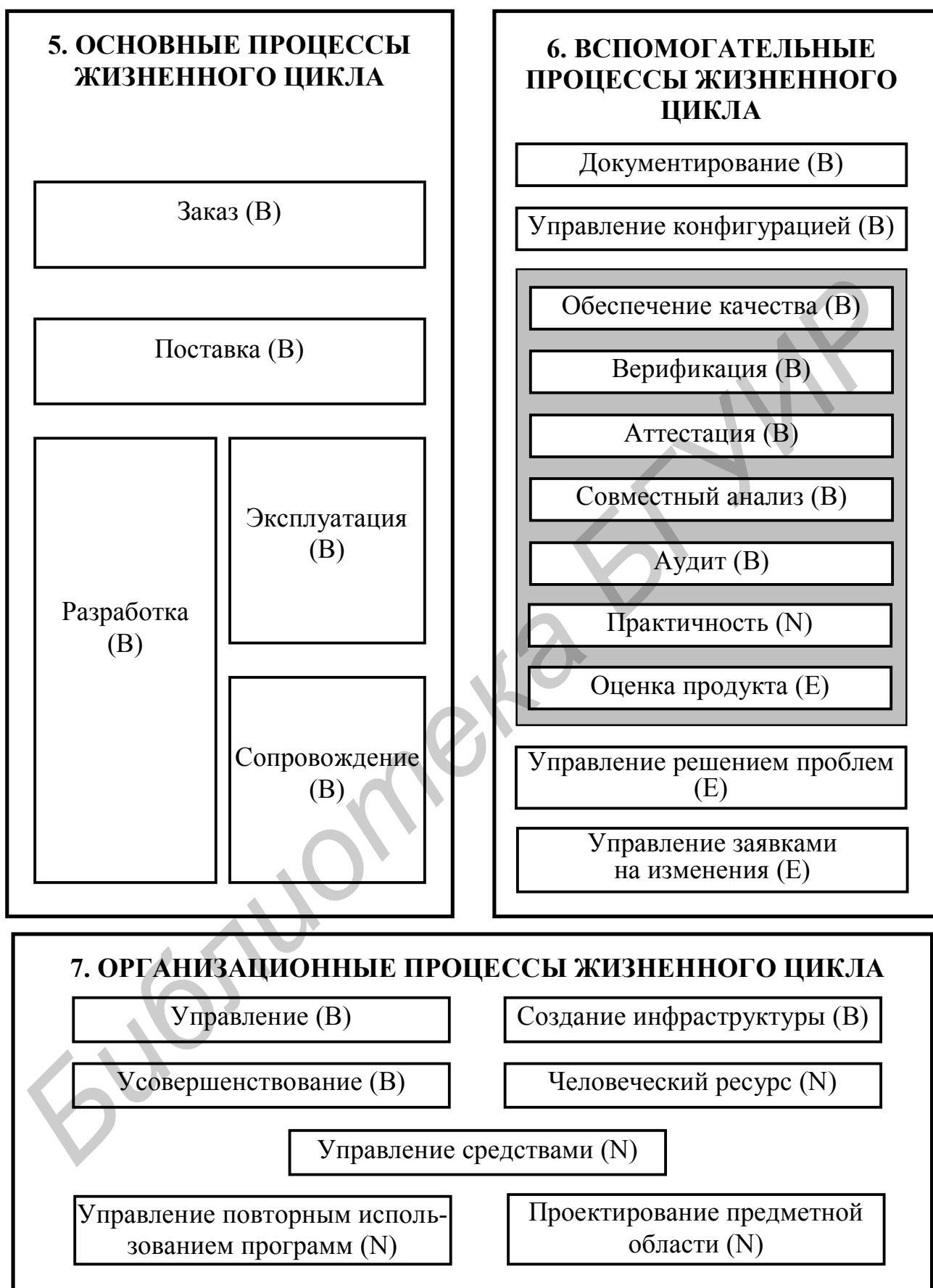


Рис. 28. Процессы жизненного цикла программных средств с учетом положений ISO/IEC 12207:1995/Amd.1:2002 и ISO/IEC 12207:1995/Amd.2:2004

*Приложение F* Дополнения *Amd.1:2002* содержит описание всех процессов и подпроцессов в виде их целей и результатов, без учета их внутренней структуры. Содержание данного приложения исправлено и расширено в Дополнении *Amd.2:2004*.

*Приложение H Amd.1:2002* содержит расширенное описание процесса заказа, согласованное с положениями серии стандартов *ISO/IEC 15504* [30– 37], которое может использоваться вместо описаний процесса заказа из *ISO/IEC 12207* и Приложения F *Amd.1:2002*.

## **3.2. Новые процессы жизненного цикла программных средств**

### **3.2.1. Процесс практичности (The Usability Process)**

Процесс практичности относится к вспомогательным процессам жизненного цикла программных средств.

**Процесс практичности** определяет работы и задачи специалиста по практичности. Процесс содержит работы, учитывающие интересы и нужды людей, которые будут связаны с данным программным средством (системой) в процессе его разработки или эксплуатации. Процесс практичности предназначен для обеспечения качества в использовании программных средств (см. подразд. 10.4 пособия).

Цели и результаты данного процесса должны удовлетворять требованиям стандартов *ISO 13407:1999*, *ISO/IEC 9126–1:2001* и *ISO/IEC TR 9126–4:2004* [20, 49, 52]. Процессу практичности по отношению к структуре стандарта *ИСО/МЭК 12207* присвоен иерархический номер 6.9.

Процесс практичности состоит из *трех работ*:

6.9.1. Подготовка процесса.

6.9.2. Проектирование, ориентированное на человека.

6.9.3. Человеческие аспекты стратегии, внедрения и обслуживания.

В данном списке и далее в разделе работы пронумерованы в соответствии с их иерархическим положением по отношению к структуре стандарта *ИСО/МЭК 12207*.

Общее число задач по данным работам равно 13.

### **3.2.2. Процесс человеческого ресурса (The Human Resource Process)**

Процесс обучения, определенный в стандарте *ISO/IEC 12207:1995*, в Дополнении *ISO/IEC 12207:1995/Amd.1:2002* [18] включен в состав процесса чело-

веческого ресурса. Данный процесс относится к группе организационных процессов жизненного цикла программных средств.

**Процесс человеческого ресурса** предназначен для обеспечения организации и проектов персоналом, обладающим соответствующими навыками и знаниями для эффективного выполнения своих ролей и совместной работы в составе группы. Данному процессу по отношению к структуре стандарта *ИСО/МЭК 12207* присвоен иерархический номер 7.4.

Процесс человеческого ресурса состоит из *шести работ*:

- 7.4.1. Подготовка процесса.
- 7.4.2. Определение требований к обучению.
- 7.4.3. Набор квалифицированного персонала.
- 7.4.4. Оценка персонала.
- 7.4.5. Установка требований к коллективу проекта.
- 7.4.6. Управление знаниями.

Общее число задач по данным работам равно 15.

Соответствующие стороны работ процесса человеческого ресурса с учетом их целей сгруппированы в три подпроцесса (рис. 29):

- управление человеческим ресурсом (The Human Resource Management);
- обучение (The Training);
- управление знаниями (The Knowledge Management).



Рис. 29. Подпроцессы процесса человеческого ресурса в соответствии с ISO/IEC 12207:1995 / Amd.1:2002

Нумерация процесса и подпроцессов, начинающаяся с буквы F, здесь и на последующих рисунках данного раздела соответствует их нумерации в *Приложении F* Дополнения *Amd.1:2002*. Связано это с отсутствием иерархического места для подпроцессов в структуре действующего стандарта *ИСО/МЭК 12207*.

Подпроцессы управления человеческим ресурсом и управления знаниями (см. рис. 29) относятся к типу новых подпроцессов жизненного цикла программных средств.

Цели и результаты выполнения процесса человеческого ресурса должны удовлетворять требованиям стандарта *ISO/IEC 15504* [30 – 37].

### **3.2.3. Процесс управления средствами (The Asset Management Process)**

Данный процесс относится к группе организационных процессов жизненного цикла программных средств.

**Процесс управления средствами** определяет работы и задачи администратора средств. Данный процесс предназначен для обеспечения возможности многократного использования средств, имеющихся в организации. В процессе используются специальные административные и технические процедуры на протяжении жизненного цикла средств с целью:

- идентификации, сертификации и определения базовой линии средств;
- контроля за модификациями, перемещениями и новыми версиями средств;
- создания отчета о состоянии этих средств;
- управления хранением и доставкой средств для повторного использования и их изъятием.

Данному процессу по отношению к структуре стандарта *ИСО/МЭК 12207* присвоен иерархический номер 7.5.

Процесс управления средствами состоит из *трех работ*:

7.5.1. Подготовка процесса.

7.5.2. Определение хранения и передачи средств.

7.5.3. Руководство и управление средствами.

Общее число задач по данным работам равно 15.

Цели и результаты выполнения процесса управления средствами должны удовлетворять требованиям стандарта *IEEE 1517–1999* [15].

### **3.2.4. Процесс управления повторным использованием программ (The Reuse Program Management Process)**

Данный процесс относится к группе организационных процессов жизненного цикла программных средств.

**Процесс управления повторным использованием программ** определяет работы и задачи администратора повторного использования программ. Данный процесс используется для планирования, установки, руководства, управления и надзора за повторным использованием программ, имеющихся в организации. Данному процессу по отношению к структуре стандарта *ИСО/МЭК 12207* присвоен иерархический номер 7.6.

Процесс управления повторным использованием программ состоит из *шести работ*:

7.6.1. Инициализация.

7.6.2. Определение предметной области (области знаний).

7.6.3. Оценка повторного использования.

7.6.4. Планирование.

7.6.5. Выполнение и управление.

7.6.6. Анализ и оценка.

Общее число задач по данным работам равно 24.

Цели и результаты процесса управления повторным использованием программ должны удовлетворять требованиям стандарта *IEEE 1517–1999* [15].

### **3.2.5. Процесс проектирования предметной области (The Domain Engineering Process)**

Данный процесс относится к группе организационных процессов жизненного цикла программных средств.

*Процесс проектирования предметной области* определяет работы и задачи инженера – специалиста в предметной области. Процесс охватывает разработку и сопровождение моделей и архитектуры предметной области, а также средств предметной области, предназначенных для объединения разрабатываемых в проекте ПС. Данному процессу по отношению к структуре стандарта *ИСО/МЭК 12207* присвоен иерархический номер 7.7.

Процесс проектирования предметной области состоит из *пяти работ*:

7.7.1. Подготовка процесса.

7.7.2. Анализ предметной области.

7.7.3. Проектирование предметной области.

7.7.4. Обеспечение средствами.

7.7.5. Сопровождение средств.

Общее число задач по данным работам равно 27.

Цели и результаты выполнения процесса проектирования предметной области должны удовлетворять требованиям стандарта *IEEE 1517–1999* [15].

Работы 7.7.2 и 7.7.3 могут быть автоматизированы за счет применения инструментальных средств, поддерживающих разработку моделей предметной области. К универсальным инструментальным средствам, автоматизирующим построение различных видов моделей, можно отнести семейство *Telelogic TAU* (см. п. 2.6.2 пособия). Из данного семейства процессу проектирования предметной области наиболее соответствует *TAU/Model Author*. Разработку функциональной модели предметной области поддерживает CASE-средство *BPwin*. Разработку информационной модели предметной области поддерживает CASE-средство *Erwin*.

### **3.3. Расширенные процессы жизненного цикла программных средств**

#### **3.3.1. Процесс оценки продукта (The Product Evaluation Process)**

*Процесс оценки продукта* является расширением и выделением в отдельный вспомогательный процесс всех работ, связанных с оценкой характеристик продукта. Целью данного процесса является обеспечение гарантий, достигаемых за счет систематических проверок и измерений, подтверждающих, что продукт соответствует заявленным и подразумеваемым потребностям пользователя. Процесс оценки продукта относится к вспомогательным процессам жизненного цикла программных средств. Данный процесс по отношению к структуре стандарта *ИСО/МЭК 12207* имеет иерархический номер 6.10.

При выполнении процесса оценки продукта:

- устанавливаются требования к оценке;
- определяются критерии оценки продукта;
- определяются методы оценки и необходимые для этого работы;
- накапливаются результаты измерений, выполняется их оценка в соответствии с установленными критериями;
- результаты работ по оценке доводятся до сведения заинтересованных сторон.

Цели и результаты данного процесса должны удовлетворять требованиям серии стандартов *ISO/IEC 14598* [21 – 26].

Существуют инструментальные средства, позволяющие автоматизировать процесс оценки продукта при выполнении различных работ процесса разработки. Например, при оценке результатов подпроцесса построения программных средств (см. рис. 32 пособия) может быть использовано семейство инструментальных средств *Telelogic TAU Logiscope* (см. п. 2.6.5 пособия).

#### **3.3.2. Процесс управления решением проблем (The Problem Resolution Management Process)**

*Процесс управления решением проблем* является расширением процесса решения проблем из базового стандарта *ИСО/МЭК 12207* (см. рис. 3 и 28). Целью данного процесса является обеспечение гарантий того, что выполнены идентификация, анализ и управление всеми обнаруженными проблемами и проконтролировано их решение. Данный процесс по отношению к структуре стандарта *ИСО/МЭК 12207* имеет иерархический номер 6.8.

При выполнении процесса управления решением проблем:



- разрабатывается стратегия управления проблемами;
- выполняется регистрация, идентификация и классификация проблем;
- выполняется анализ и оценка проблем с целью выбора соответствующих решений;
- реализуется решение проблем;
- выполняется отслеживание проблем до их устранения;
- фиксируется текущее состояние всех проблем.

Процесс управления решением проблем может инициировать заявки на изменения (см. п. 3.3.3 пособия).

### **3.3.3. Процесс управления заявками на изменения (The Change Request Management Process)**

*Процесс управления заявками на изменения* является расширением и выделением в отдельный вспомогательный процесс всех работ, связанных с заявками на изменения и соответствующими изменениями промежуточных или конечных продуктов проекта. Целью данного процесса является обеспечение гарантий того, что заявки на изменения управляются, отслеживаются и контролируются. Данный процесс по отношению к структуре стандарта *ИСО/МЭК 12207* имеет иерархический номер 6.11.

При выполнении процесса управления заявками на изменения:

- разрабатывается стратегия управления изменениями;
- регистрируются и идентифицируются заявки на изменения;
- идентифицируются зависимости и связи между заявками на изменения;
- определяются критерии для санкционирования реализации заявок на изменения;
- утверждаются и располагаются в соответствии с приоритетом заявки на изменения, оцениваются требования к необходимым для их реализации ресурсам;
- инициируются изменения на основе их приоритетов и доступности ресурсов;
- реализуются и отслеживаются до завершения утвержденные изменения;
- фиксируется текущее состояние всех изменений.

Для повышения эффективности процесса управления заявками на изменения следует автоматизировать его выполнение, используя соответствующие инструментальные средства. К современным инструментальным средствам, предназначенным для автоматизации управления изменениями, относится *Telelogic SYNERGY* (см. п. 2.6.3 пособия). Такая автоматизация позволит существенно повысить как качество процессов разработки и сопровождения, так и качество программных продуктов.

### 3.4. Процессы жизненного цикла с измененной структурой

В Дополнениях *Amd.1:2002* и *Amd.2:2004* [18, 19] основным изменениям подверглась структура процессов заказа, поставки, разработки, эксплуатации и управления.

#### 3.4.1. Процесс заказа (The Acquisition Process)

Базовая структура процесса заказа (номер 5.1 в стандарте *ИСО/МЭК 12207*) рассмотрена в п. 2.2.1 пособия и представлена на рис. 5. Модифицированная структура процесса заказа приведена на рис. 30.

Из рис. 30 видно, что в структуру процесса заказа введены четыре составных подпроцесса:

- подготовка заказа (The Acquisition Preparation);
- выбор поставщика (The Supplier Selection);
- надзор за поставщиком (The Supplier Monitoring);
- приемка договора (The Customer Acceptance).

Каждый из подпроцессов группирует по целевому признаку и выходным результатам соответствующие стороны работ процесса заказа.

Цели и результаты выполнения процесса заказа и его подпроцессов должны удовлетворять требованиям стандарта *ISO/IEC 15504* [30 – 37].

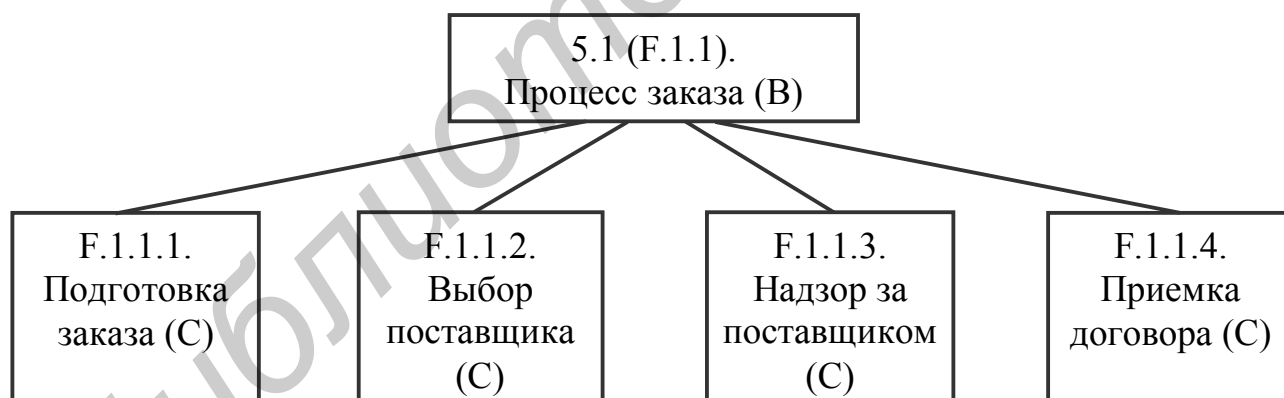


Рис. 30. Подпроцессы процесса заказа  
в соответствии с *ISO/IEC 12207:1995 / Amd.1:2002*

#### 3.4.2. Процесс поставки (The Supply Process)

Базовая структура процесса поставки (номер 5.2 в стандарте *ИСО/МЭК 12207*) рассмотрена в п. 2.2.2 пособия и представлена на рис. 6. Модифицированная структура процесса поставки приведена на рис. 31.

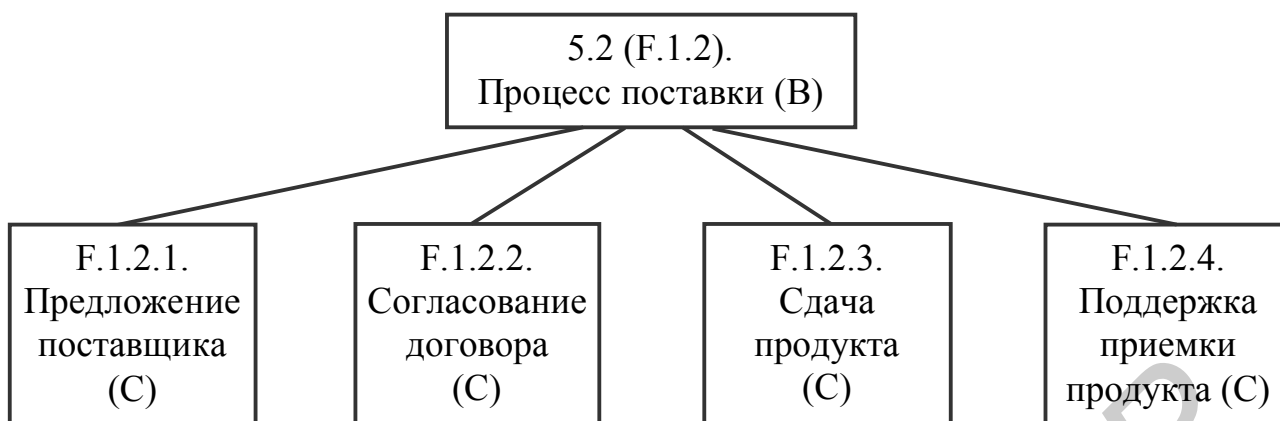


Рис. 31. Подпроцессы процесса поставки  
в соответствии с ISO/IEC 12207:1995 / Amd.2:2004

Из рис. 31 видно, что в структуру процесса поставки введены четыре составных подпроцесса:

- предложение поставщика (The Supplier Tendering);
- согласование договора (The Contract Agreement);
- сдача продукта (The Product Release);
- поддержка приемки продукта (The Product Acceptance Support).

Каждый из подпроцессов группирует по целевому признаку и выходным результатам соответствующие стороны работ процесса поставки.

### 3.4.3. Процесс разработки (The Development Process)

Базовая структура процесса разработки (номер 5.3 в стандарте *ИСО/МЭК 12207*) рассмотрена в п. 2.2.3 пособия и представлена на рис. 7. Модифицированная структура процесса разработки приведена на рис. 32.

В отличие от процесса разработки стандарта *ИСО/МЭК 12207*, состоящего из тринадцати работ, процесс разработки в соответствии с Дополнением *ISO/IEC 12207:1995/Amd.1:2002* представляет собой совокупность из одиннадцати базовых (B), расширенных (E) и составных (C) подпроцессов.

Основные отличия модифицированной структуры процесса разработки (см. рис. 32) от его базовой структуры (см. рис. 7) заключаются в следующем.

1. Работа 5.3.1 (Подготовка процесса разработки) заменена на расширенный подпроцесс *F.1.3.1 выявления требований (The Requirements Elicitation)*. Данный подпроцесс предназначен для сбора и обработки требований заказчика, а также отслеживания изменений в них.

При выполнении данного подпроцесса формируется базовая линия согласованных с заказчиком требований, что является основой определения необходимых рабочих продуктов. Устанавливается механизм оценки изменений в тре-

бования заказчика и их преобразования в базовую линию требований. Устанавливается механизм, позволяющий заказчику легко определить состав и взаимоотношение своих требований. Определяется механизм надзора за потребностями заказчика.

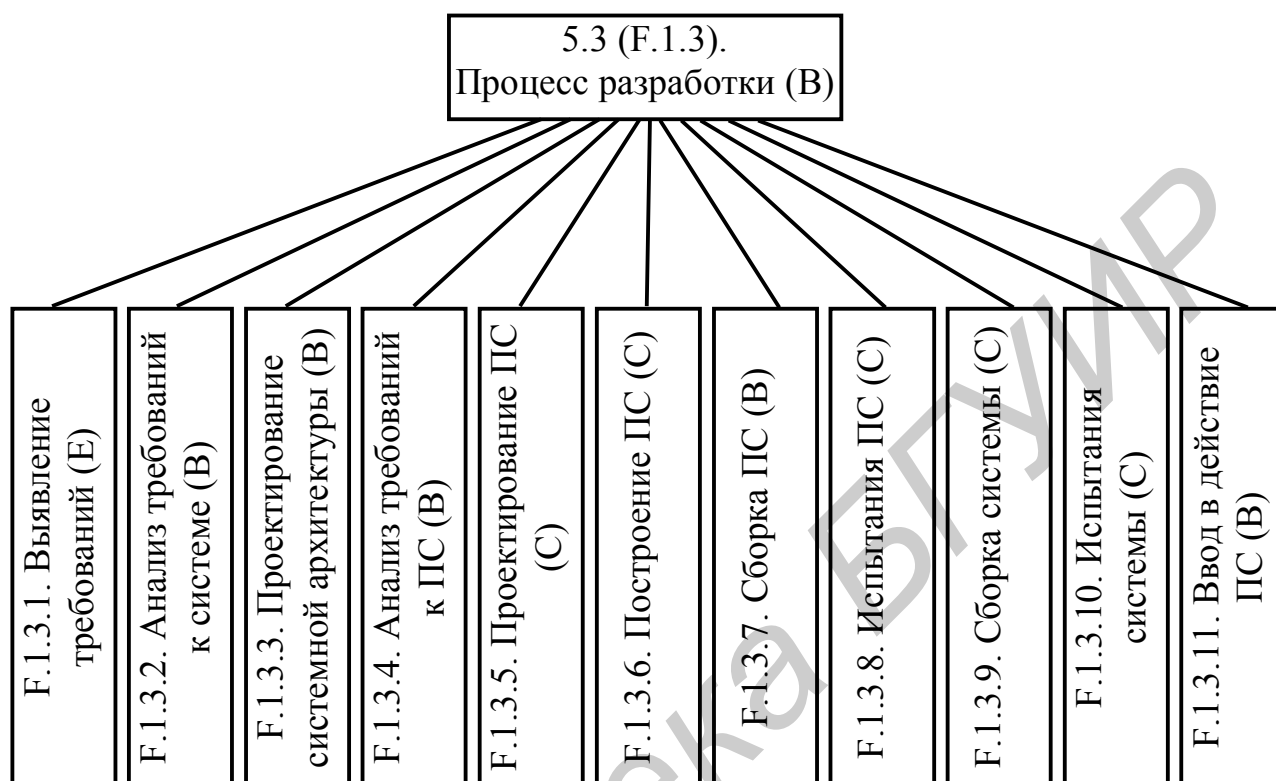


Рис. 32. Структура процесса разработки  
в соответствии с ISO/IEC 12207:1995 / Amd.1:2002

2. Работы 5.3.5 (Проектирование программной архитектуры) и 5.3.6 (Техническое проектирование программных средств) объединены в *подпроцесс F.1.3.5 проектирования программных средств (The Software Design)*. Данный подпроцесс предназначен для разработки проекта программного средства и предусматривает возможность верификации результатов проектирования на соответствие требованиям.

При выполнении данного подпроцесса выполняется проектирование программной архитектуры, в ходе которого определяются компоненты программного средства, реализующие установленные требования к нему. Определяются внешние и внутренние интерфейсы для каждого из компонентов. Выполняется техническое проектирование, в котором определяются программные модули, готовые к дальнейшему кодированию и тестированию. Устанавливается соответствие требований к программному средству и результатов проектирования.

3. Работа 5.3.7 (Программирование и тестирование программных средств) заменена *подпроцессом F.1.3.6 построения программных средств (The Software Construction)*. Данный подпроцесс предназначен для реализации исполняемых

программных модулей, соответствующих результатам проектирования программных средств.

При выполнении данного подпроцесса определяются критерии верификации программных модулей на соответствие требованиям к ним. Каждый модуль кодируется. Устанавливается соответствие между требованиями к программному средству, результатами проектирования и программными модулями. Выполняется верификация программных модулей на соответствие требованиям.

4. Работа 5.3.9 (Квалификационные испытания программных средств) заменена *подпроцессом F.1.3.8 испытаний программных средств (The Software Testing)*. Данный подпроцесс предназначен для подтверждения того, что программный продукт, полученный в результате сборки, соответствует установленным к нему требованиям.

При выполнении данного подпроцесса устанавливаются критерии оценки соответствия собранного программного продукта требованиям. С использованием установленных критериев выполняется верификация программного продукта. Разрабатывается стратегия регрессионного тестирования, применяемая для повторного тестирования программного продукта после внесения изменений в его элементы.

5. Работа 5.3.11 (Квалификационные испытания системы) заменена *подпроцессом F.1.3.10 испытаний системы (The System Testing)*. Данный подпроцесс предназначен для того, чтобы удостовериться, что реализация каждого требования к системе протестирована на соответствие и вся система готова к общему заключению о соответствии.

При выполнении данного подпроцесса устанавливаются критерии для оценки соответствия собранной системы системным требованиям. Собранная система верифицируется, используя установленные требования. Разрабатывается стратегия регрессионного тестирования системы. Данная стратегия применяется для повторного тестирования системы после внесения в него изменений.

6. Работа 5.3.13 (Обеспечение приемки программных средств) перенесена в процесс поставки (см. подпроцесс поддержки приемки продукта на рис. 31).

Цели и результаты выполнения процесса разработки должны удовлетворять требованиям стандарта *ISO/IEC 15504* [30 – 37].

### **3.4.4. Процесс эксплуатации (The Operation Process)**

Базовая структура процесса эксплуатации (номер 5.4 в стандарте *ISO/МЭК 12207*) рассмотрена в п. 2.2.4 пособия и представлена на рис. 11. Модифицированная структура процесса эксплуатации приведена на рис. 33. В структуру процесса эксплуатации введены два расширенных подпроцесса:

- эксплуатационное использование (The Operational Use);
- поддержка пользователя (The Customer Support).

Каждый из подпроцессов группирует соответствующие работы процесса эксплуатации в соответствии с их целью и выходными результатами.

Цели и результаты выполнения процесса эксплуатации и его подпроцессов должны удовлетворять требованиям стандарта *ISO/IEC 15504* [30 – 37].

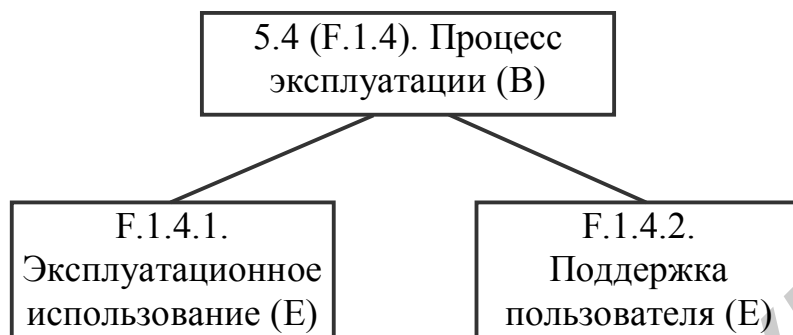


Рис. 33. Подпроцессы процесса эксплуатации в соответствии с *ISO/IEC 12207:1995 / Amd.1:2002*

### 3.4.5. Процесс управления (The Management Process)

Модифицированная структура процесса управления приведена на рис. 34. В отличие от базового процесса управления (см. рис. 23) модифицированный процесс разделяется на шесть подпроцессов, связанных с управлением отдельными сторонами жизненного цикла программных средств. Большинство из данных подпроцессов представляет собой объединение по целевому признаку существующих в стандарте *ISO/IEC 12207:1995* работ и задач ЖЦ с учетом их необходимого развития и уточнения (Е).

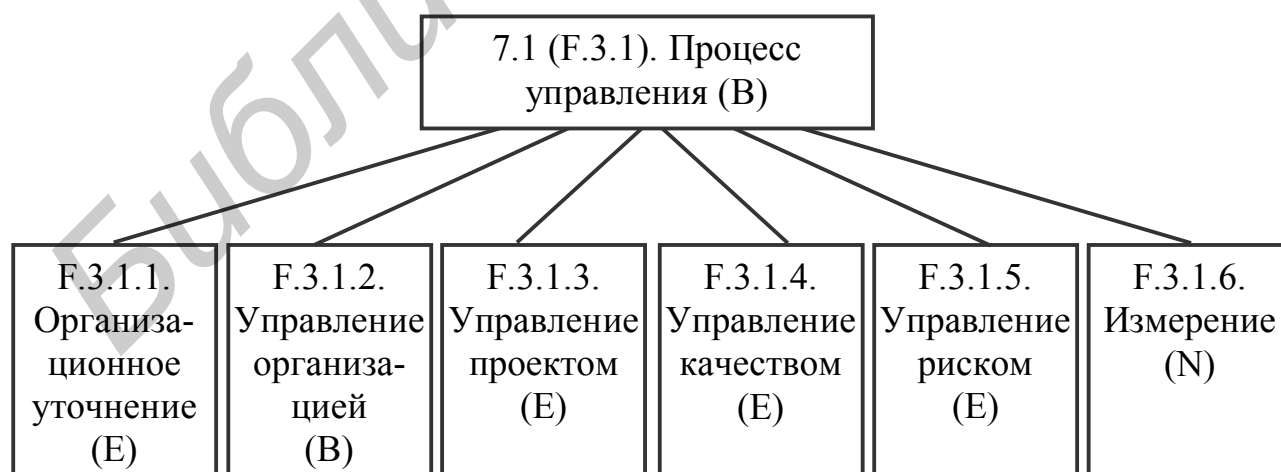


Рис. 34. Структура процесса управления в соответствии с *ISO/IEC 12207:1995 / Amd.1:2002*

*Подпроцесс организационного уточнения (The Organizational Alignment)* предназначен для приспособливания процессов ЖЦ ПС к деловым целям организации. При выполнении данного подпроцесса идентифицируются деловые цели организации и структура процессов ЖЦ ПС, формируется общая стратегия определения, выполнения и усовершенствования процессов, определяются роли сотрудников в достижении деловых целей организации.

*Подпроцесс управления организацией (The Organization Management)* предназначен для установки и реализации в организации методов управления процессами ЖЦ ПС, совместимых с деловыми целями организации.

В результате выполнения данного подпроцесса организация вкладывает денежные средства в целесообразную инфраструктуру управления, определяются лучшие методы поддержки эффективной организации и управления проектами, обеспечивается основа для достижения деловых целей организации, базирующаяся на данных методах управления.

*Подпроцесс управления проектом (The Project Management)* предназначен для идентификации, установки, координации и контроля работ, задач и ресурсов, необходимых для создания продукта или услуги в соответствии с требованиями и ограничениями проекта.

В результате выполнения подпроцесса определяется область работы над проектом; оценивается достижимость целей проекта при доступных ресурсах и ограничениях; подбираются и предварительно оцениваются задачи и ресурсы, необходимые для завершения работы; определяются и контролируются взаимосвязи между элементами проекта, проектами и организационными элементами; разрабатываются и реализуются планы выполнения проекта; контролируется ход выполнения проекта; предпринимаются действия для коррекции отклонений от плана и предотвращения повторения возникших проблем.

*Подпроцесс управления качеством (The Quality Management)* выполняется путем постоянного контроля качества продуктов и услуг на организационном и проектном уровнях с целью обеспечения гарантии их соответствия исходным требованиям.

В результате выполнения подпроцесса устанавливаются цели по уровню качества, основанные на явных и неявных требованиях заказчика к качеству; разрабатывается общая стратегия достижения установленных целей; для реализации стратегии определяется система управления качеством; выполняются установленные работы по управлению качеством и обеспечению качества; контролируется достижение целей по уровню качества; предпринимаются соответствующие действия, если цели по уровню качества не достигаются.

*Подпроцесс управления риском (The Risk Management)* предназначен для постоянного выявления, управления и предотвращения рисков как на организационном, так и на проектном уровнях. В результате выполнения подпроцесса определяется область управления рисками; определяются и реализуются стратегии управления рисками; риски анализируются и устанавливаются их приоритеты; выбираются методы отслеживания рисков; выполняются действия для предотвращения или сокращения воздействия рисков.

В стандарте *ISO/IEC 16085:2004* [42] содержатся требования и рекомендации по выполнению подпроцесса управления риском.

*Подпроцесс измерения (The Measurement)* является новым подпроцессом процесса управления. Подпроцесс предназначен для сбора и анализа данных, связанных с разрабатываемыми продуктами или реализуемыми процессами внутри организации или ее проектов и используемых для поддержки эффективного управления процессами и объективной демонстрации качества программных продуктов.

Подпроцесс измерения содержит *четыре задачи*:

- установка и сопровождение набора измерений;
- планирование процесса измерения;
- выполнение измерения в соответствии с планом;
- оценка измерения.

Стандарт *ISO/IEC 15939:2002* содержит требования и рекомендации по выполнению подпроцесса измерения [41].

Цели и результаты выполнения процесса управления должны удовлетворять требованиям серии стандартов *ISO/IEC 15504* [30 – 37].

### **3.4.6. Расширенное описание процесса заказа**

В *Приложении H* Дополнения *ISO/IEC 12207:1995/Amd.1:2002* дано расширенное описание процесса заказа в терминах стандарта *ISO/IEC 12207:1995* [17], согласованное с требованиями стандарта *ISO/IEC 15504* [30 – 37]. Данное описание может использоваться вместо описаний процесса заказа, приведенных в стандарте *ISO/IEC 12207:1995* и в *Приложении F* Дополнения *Amd.1:2002*. Оно призвано обеспечить возможность оценки процесса заказа и повысить степень выявления рисков при приобретении системы или программного средства.

К расширенному процессу заказа относятся *семнадцать подпроцессов*:

- 1) политика заказа (The Acquisition Policy);
- 2) стратегия заказа (The Acquisition Strategy);
- 3) анализ суммарных выгод (The Benefits Analysis);
- 4) технические требования (The Technical Requirements);
- 5) правовые и административные требования (The Legal and Administrative Requirements);
- 6) финансовые требования (The Financial Requirements);
- 7) требования к проекту (The Project Requirements);
- 8) заявка на подряд (тендер) (The Request for Proposals);
- 9) выбор поставщика (The Supplier Qualification);
- 10) предварительная оценка (The Proposal Evaluation);
- 11) согласование договора (The Contract Agreement);
- 12) надзор за поставщиком (The Supplier Monitoring);
- 13) приемка договора (The Acceptance);
- 14) закрытие договора (The Contract Closure);



- 15) связи с поставщиком (The Supplier Relationships);
- 16) связи с пользователем (The User Relationships);
- 17) финансовое руководство (The Financial Management).

Данные подпроцессы объединяют по целям и результатам соответствующие стороны работ и задач процесса заказа.

Структура расширенного процесса заказа на уровне работ приведена на рис. 35. В состав расширенного процесса заказа, помимо описанных в *ISO/IEC 12207:1995* (см. рис. 5), добавлено *пять работ*:

- 5.1.6. Закрытие договора.
- 5.1.7. Политика заказа.
- 5.1.8. Управление связями с поставщиком.
- 5.1.9. Управление связями с пользователем.
- 5.1.10. Финансовое руководство.

Нумерация работ в данном списке соответствует их иерархическому положению по отношению к структуре стандарта *ИСО/МЭК 12207*.

В состав новой работы 5.1.6, название которой перекликается со старой работой 5.1.5, вошли задачи по закрытию договора, которые не нашли отражения в работе 5.1.5.

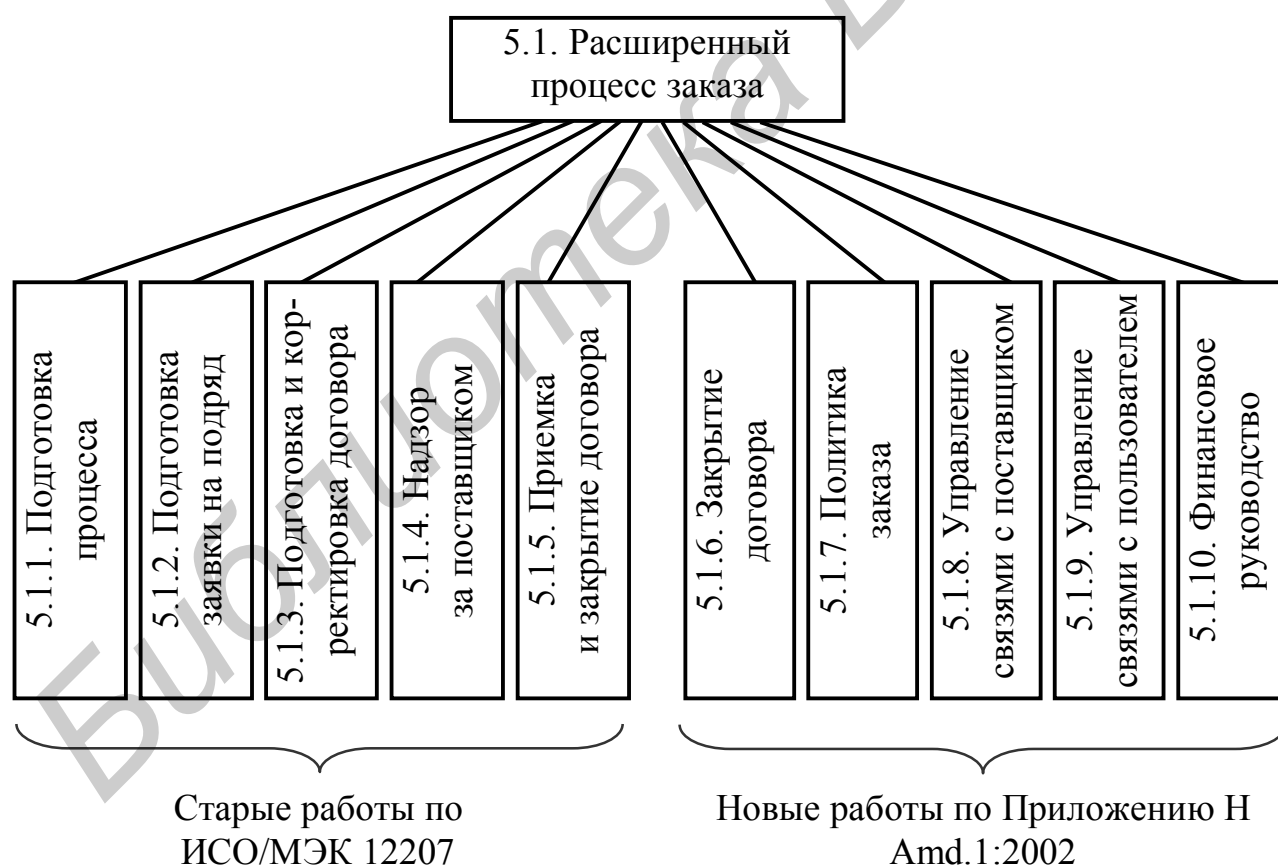


Рис. 35. Работы расширенного процесса заказа в соответствии с Приложением Н ISO/IEC 12207:1995 / Amd.1:2002

## 4. СТАНДАРТ ISO/IEC TR 15271

### 4.1. Общие сведения

В 1998 г. разработан международный стандарт *ISO/IEC TR 15271:1998 – Информационная технология – Руководство по применению ISO/IEC 12207 (Процессы жизненного цикла программных средств)* [28]. В России аутентичный стандарт *ГОСТ Р ИСО/МЭК ТО 15271–2002* действует с 2003 г. [8].

Данный стандарт дополняет стандарт *ИСО/МЭК 12207*. В нем приведены рекомендации по практическому применению стандарта *ИСО/МЭК 12207* при реализации конкретных проектов создания программных средств.

Стандарт имеет следующую *структуру*:

1. Область применения.
2. Нормативные ссылки.
3. Система обозначений.
4. Основные концепции в развитие ИСО/МЭК 12207.
5. Внедрение ИСО/МЭК 12207.
6. Применение в проектах.
7. Применение в организациях.
8. Прикладное применение модели жизненного цикла в организациях.

Приложение А. Процессы качества и требования к оценке.

Приложение В. Классификация выходных результатов процессов.

Приложение С. Модели жизненного цикла.

Приложение D. Примеры адаптации ИСО/МЭК 12207.

В данном разделе пособия приведены основные положения стандарта *ИСО/МЭК ТО 15271*.

### 4.2. Основные концепции стандарта ИСО/МЭК 12207

Стандарт *ИСО/МЭК 12207* устанавливает *архитектуру верхнего уровня жизненного цикла программных средств* от замысла до снятия с эксплуатации (утилизации). Архитектура состоит из процессов и связей между ними. Процессы основаны на *двух принципах: модульности и ответственности*.

Процессы являются *модульными*, поскольку для них характерны:

- высокая связность (все части процесса строго взаимоувязаны);
- слабое сцепление (число интерфейсов между процессами минимально).

Принцип *ответственности* означает, что каждый процесс рассмотрен с точки зрения ответственности (обязанностей) стороны, его выполняющей.

Набор процессов, приведенный в стандарте *ИСО/МЭК 12207*, используется для больших или сложных программных проектов. Однако стандарт *ИСО/МЭК 12207* после соответствующей адаптации может быть применен к программному проекту любого типа, меньшего размера и сложности. Данный стандарт может использоваться для ПС, являющихся как самостоятельными объектами, так и частями системы.

Если ПС является частью системы, его выделяют из системы, разрабатывают и включают в систему. Хотя стандарт *ИСО/МЭК 12207* охватывает жизненный цикл системы в целом, он *определяет процессы разработки, эксплуатации и сопровождения системы только в части ее программных средств*.

В процессе разработки программного средства различают **два вида работ**: системные и программные. *Системные работы* начинают и завершают процесс разработки. К данным работам относятся следующие (см. рис. 7).

5.3.2. Анализ требований к системе.

5.3.3. Проектирование системной архитектуры.

5.3.10. Сборка системы.

5.3.11. Квалификационные испытания системы.

В работе 5.3.2 разрабатываются системные требования. В работе 5.3.3 выполняется разработка архитектурного (эскизного) проекта системы, в котором осуществляется разделение системы на технические средства, программные средства и ручные операции.

Работы с 5.3.4 (анализ требований к программным средствам) по 5.3.9 (квалификационные испытания программных средств) представляют собой *программные работы*. Они выполняются над выделенными из системы программными средствами.

После разработки ПС в работе 5.3.10 их интегрируют с техническими средствами системы и ручными операциями. Системные работы завершаются квалификационными испытаниями системы (работа 5.3.11). Таким образом, системные работы являются расширением набора программных работ.

Управление каждым из основных и вспомогательных процессов ЖЦ ПС (см. рис. 3 пособия) на проектном уровне реализуют с помощью **процесса управления** (см. рис. 23 пособия). В процессе управления осуществляют планирование проекта, реализацию и контроль всех запланированных действий.

План управления проектом должен быть подготовлен в процессе поставки (см. задачу 5.2.4.5 в п. 2.2.2 пособия). В зависимости от объема и сложности проекта отдельные разделы плана управления (см. п. 2.2.2) могут быть подготовлены в виде отдельных (дополнительных) планов.

Сводные сведения о наборе дополнительных планов могут быть получены из таблиц, приведенных в разд. 2 учебного пособия, или из *Приложения В* к стандарту *ИСО/МЭК ТО 15271* [8, 28].

Подробно процесс управления проектом описан в стандарте *ISO/IEC TR 16326:1999* [43] (аутентичном стандарте России *ГОСТ Р ИСО/МЭК ТО 16326–2002* [9]) и поясняется в разд. 5 пособия.

Планирование **управления документами** определено в процессе докумен-

тирования *ИСО/МЭК 12207* (см. п. 2.3.1 пособия) и подробно пояснено в стандарте *ISO/IEC 15910:1999* [40] и аутентичном стандарте России *ГОСТ Р ИСО/МЭК 15910–2002* [7]. Данные стандарты рассмотрены в разд. 6 пособия.

В *Приложении В* стандарта *ИСО/МЭК ТО 15271* содержатся классифицированные по процессам сводные таблицы основных выходных результатов задач, которые должны быть документально оформлены, с указанием типа документа.

*Основными типами документов ЖЦ* являются:

- спецификация;
- план;
- описание;
- отчет;
- процедура;
- протокол;
- программный продукт (исходные коды).

*Принципы управления качеством* реализованы в *ИСО/МЭК 12207* тремя основными способами:

- *интеграция качества в жизненный цикл*; стандарт *ИСО/МЭК 12207* обеспечивает для каждого процесса ЖЦ возможность выполнения цикла «план – реализация – контроль – действие» (Plan – Do – Check – Act, PDCA) посредством процесса усовершенствования; при этом все работы, связанные с качеством, входят в соответствующие процессы ЖЦ;

- *процесс обеспечения качества*; данный процесс предназначен для обеспечения соответствия ПП и программной услуги требованиям и планам; лица, ответственные за данный процесс, должны быть наделены организационной независимостью и соответствующими полномочиями;

- *процесс усовершенствования*; данный процесс служит для дальнейшего повышения качества работ организации в целом независимо от конкретных проектов.

*Стандарт ИСО/МЭК 12207* является *гибким и учитывает развитие технологий* за счет обеспечения открытой архитектуры высокого уровня. Данный стандарт:

- применим к любым моделям ЖЦ, методам и технологиям программной инженерии (объектно-ориентированное или структурное проектирование, нисходящее проектирование и тестирование, макетирование), языкам программирования;

- гибок, поскольку работы и задачи процессов ЖЦ отвечают на вопрос «что делать», а не «как делать»;

- адаптируем к любой отрасли промышленности.

*Соответствие с ИСО/МЭК 12207* может быть достигнуто следующими способами:

- заказчик определяет условия договоров, набор процессов, работ и задач из *ИСО/МЭК 12207*, которым должны подчиняться поставщики;

- в программном проекте выполняется адаптация процессов, работ и за-

дач ЖЦ в соответствии с критериями договора.

Таким образом, пользователь стандарта *ИСО/МЭК 12207* должен выбрать, практически применить и скомпоновать процессы, работы и задачи ЖЦ соответственно целям и задачам организации и проекта. При практическом применении *ИСО/МЭК 12207* должны быть сохранены его архитектура, сущность и целостность, например, за счет включения элементов стандарта с пометкой «не применяется» с объяснением причины такого решения.

### 4.3. Внедрение ИСО/МЭК 12207

Возможны различные *способы внедрения ИСО/МЭК 12207*, в том числе:

- применение в конкретном проекте при определении процессов, работ и задач, связанных с ПС;
- использование в качестве основы для усовершенствования программных процессов в организации;
- использование в качестве компонента в процессе моделирования общего ЖЦ системы.

Можно выделить следующие *шаги стратегии практического применения* стандарта *ИСО/МЭК 12207*:

- план внедрения;
- практическое применение стандарта *ИСО/МЭК 12207*;
- проведение сопровождения пилотных проектов;
- формализация метода внедрения;
- утверждение метода внедрения.

Внедрение *ИСО/МЭК 12207* должно рассматриваться и планироваться как конкретный проект. При *планировании внедрения* должны быть определены:

- область применения проекта по внедрению (единый проект внутри организации или часть двустороннего договора; выделение одного или нескольких ключевых процессов, от которых ожидают выгоду для организации; практическое применение стандарта *ИСО/МЭК 12207* через ряд проектов с поэтапным его введением; практическое применение стандарта *ИСО/МЭК 12207* во всех проектах и во всех подразделениях организации);
- цели проекта и их соответствие целям организации;
- роли и обязанности группы персонала проекта с установлением единоличной ответственности за каждый процесс;
- ресурсы (время, деньги, персонал, оборудование), необходимые для внедрения стандарта *ИСО/МЭК 12207*;
- план управления проектом по внедрению стандарта *ИСО/МЭК 12207* (план внедрения).

*Практическое применение стандарта ИСО/МЭК 12207* должно базироваться на использовании процесса адаптации, описанного в его *Приложениях А и В* (см. подразд. 2.5 пособия), а также учитывать рекомендации стандарта *ИСО/МЭК ТО 15271* и особенности конкретной ситуации. При выполнении ра-

бот процесса адаптации должна быть выполнена привязка установленных процессов, работ и задач (с учетом их взаимосвязей) к выбранной модели (или моделям) ЖЦ ПС. Результаты адаптации и обоснование метода внедрения должны быть документированы и включены в план внедрения *ИСО/МЭК 12207*.

Для уменьшения риска внедрения стандарта *ИСО/МЭК 12207* в организации рекомендуется выполнить один или несколько **пилотных проектов**, в которых осваиваются и сопровождаются адаптированные процессы ЖЦ и определяются критические факторы достижения успеха.

**Формализация метода внедрения** включает в себя введение нового процесса в ряде проектов или в целом в организации. После **утверждения метода внедрения** *ИСО/МЭК 12207* выполняется последовательная автоматизированная реализация процесса внедрения в проекте или в организации.

## **4.4. Применение ИСО/МЭК 12207 в проектах и организациях**

В данном подразделе поясняются вопросы адаптации стандарта *ИСО/МЭК 12207*, описанные в его *Приложении В* (см. подразд. 2.5 пособия).

На практическое применение *ИСО/МЭК 12207* в конкретных проектах влияют следующие **факторы**:

- *модель ЖЦ системы* (место данного проекта в ЖЦ системы); *все ниже следующие факторы должны быть увязаны с моделью ЖЦ системы*);
- *набор политик и процедур организации* (например, связанных с защитой, безопасностью, конфиденциальностью, управлением риском, использованием независимого органа по верификации и аттестации, использованием конкретного языка программирования, обеспечением техническими ресурсами);
- *характеристики системы* (например, межсистемные и внутрисистемные интерфейсы, интерфейсы пользователя, влияние ошибок ПС на защиту и безопасность системы, оценка временных мощностей и временных ограничений, наличие реализованных техническими средствами программ, наличие соответствующих компьютеров);
- *характеристики ПС* (например, потенциальное число элементов программной конфигурации, типы, объемы и критичность ПС, технический риск, типы документов, необходимость новой разработки, модификации или повторного применения ПС, характеристики качества ПС по *ISO/IEC 9126-1-4* [49 – 52]);
- *политика сопровождения ПС* (ожидаемые период сопровождения и периодичность внесения изменений, критичность применения, персонал сопровождения и его квалификация, необходимая для сопровождения среда);
- *модель ЖЦ проекта* (например, последовательное, повторное или комбинированное выполнение работ процессов *ИСО/МЭК 12207*, структурированных в модель ЖЦ ПС);

- *участвующие в проекте стороны* (например, количество сторон, их ответственность за конкретные процессы, взаимосвязь между сторонами);
- *типы программных средств* (новая разработка; готовое ПС – целиком, с настройкой параметров под необходимую конфигурацию, модифицированное; реализованное техническими средствами; автономное; непоставляемое);
- *объем проекта* (в больших проектах, в которые вовлечены десятки или сотни лиц, необходим тщательный административный надзор и контроль с применением процессов совместного анализа, аудита, верификации, аттестации, обеспечения качества; для малых проектов такие методы контроля могут быть излишними);
- *критичность проекта* (значительная зависимость работы системы от правильного функционирования ПС и своевременности выдачи результатов; для таких ПС необходим более тщательный надзор и контроль);
- *технический риск* (например, создание уникального или сложного ПС, которое трудно сопровождать и использовать, неправильное, неточное или неполное определение требований).

В организациях стандарт *ИСО/МЭК 12207* должен использоваться в качестве составной части деятельности по усовершенствованию процессов, связанных с ПС. Применение стандарта *ИСО/МЭК 12207* в организации основано на тех же методах внедрения, что и в проектах (см. подразд. 4.3).

## 4.5. Прикладное применение модели жизненного цикла системы

*Типовая модель жизненного цикла системы* начинается с концепции потребности в системе, охватывает разработку, эксплуатацию и сопровождение системы и заканчивается снятием системы с эксплуатации (утилизацией).

Модель ЖЦ обычно разделяют на *периоды реализации*, например, стадии, этапы, фазы. Каждый период включает основные выполняемые в нем работы и задачи. При их завершении может потребоваться разрешение на переход к следующему периоду. Например, типовую модель ЖЦ системы часто разделяют на следующие *стадии (этапы)*:

- 1) определение потребностей;
- 2) исследование и описание основных концепций;
- 3) демонстрация и аттестация основных концепций;
- 4) проектирование и разработка;
- 5) создание и производство;
- 6) распространение и продажа;
- 7) эксплуатация;
- 8) сопровождение и поддержка;
- 9) снятие с эксплуатации (утилизация).

Данные этапы должны быть в дальнейшем адаптированы к конкретной модели ЖЦ системы.

**Типовая модель жизненного цикла программного средства** состоит из ряда работ. Данная модель начинается с формулировки замысла или концепции программного средства, продолжается работами по применению методов системной и программной инженерии, работами по эксплуатации, сопровождению и поддержке и заканчивается снятием ПС с эксплуатации. В стандарте *ИСО/МЭК 12207* все работы объединены в основные, вспомогательные и организационные процессы, из которых формируют модель ЖЦ ПС.

На рис. 36 приведен пример использования стандарта *ИСО/МЭК 12207* в общей модели ЖЦ гипотетической системы [8].

**На этапе определения потребностей** определяют потребности в новой или усовершенствованной системе и формулируют их с учетом стоимости, критичности, реализуемости планируемой системы.

**На этапе исследования и описания концепции** выполняется первоначальное планирование, при котором оценивают технические, стратегические и рыночно-экономические аспекты системы путем исследований, опытной разработки и оценки ее концепции. Разрабатываются ПС, реализующие аналитические или имитационные модели, необходимые при исследованиях. Выходными результатами данного этапа, передаваемыми на следующий этап, являются предварительные общие требования к системе и возможные ПС, выбранные в качестве прототипа.

**На этапе демонстрации и аттестации** уточняются характеристики системы, ее концепции и принятые решения. При этом используются методы системной инженерии. Определяются предварительные требования к ПС и его прототипам, включая тестирование и аттестацию. Характеристики системы, выбранные концепции и решения аттестуют для демонстрации того, что система пригодна для технологической разработки. Требования к системе в дальнейшем распределяются между ее компонентами, такими как технические средства, компьютеры, программные средства и персонал. Данные выходные результаты передаются на следующий этап.

**На этапе разработки** выполняется проектирование, создание, сборка, тестирование и оценка системных технических и программных средств, определение персонала подсистем и объектов сопровождения. Выходными результатами данного этапа являются система, достаточно близкая к создаваемой, необходимые документы и результаты тестирования качества системы.

На данном этапе при разработке ПС полностью применим стандарт *ИСО/МЭК 12207*. При этом возможно однократное или многократное выполнение процесса разработки. Результатами данного этапа являются исходные требования к ПС, его технический проект и тексты программ.

**На этапе изготовления** выполняются работы от постановки на производство до поставки заказчику и приемки системы. Период производства при поставке системы на рынок охватывает работы от постановки на производство до перепроектирования или снятия системы с производства.



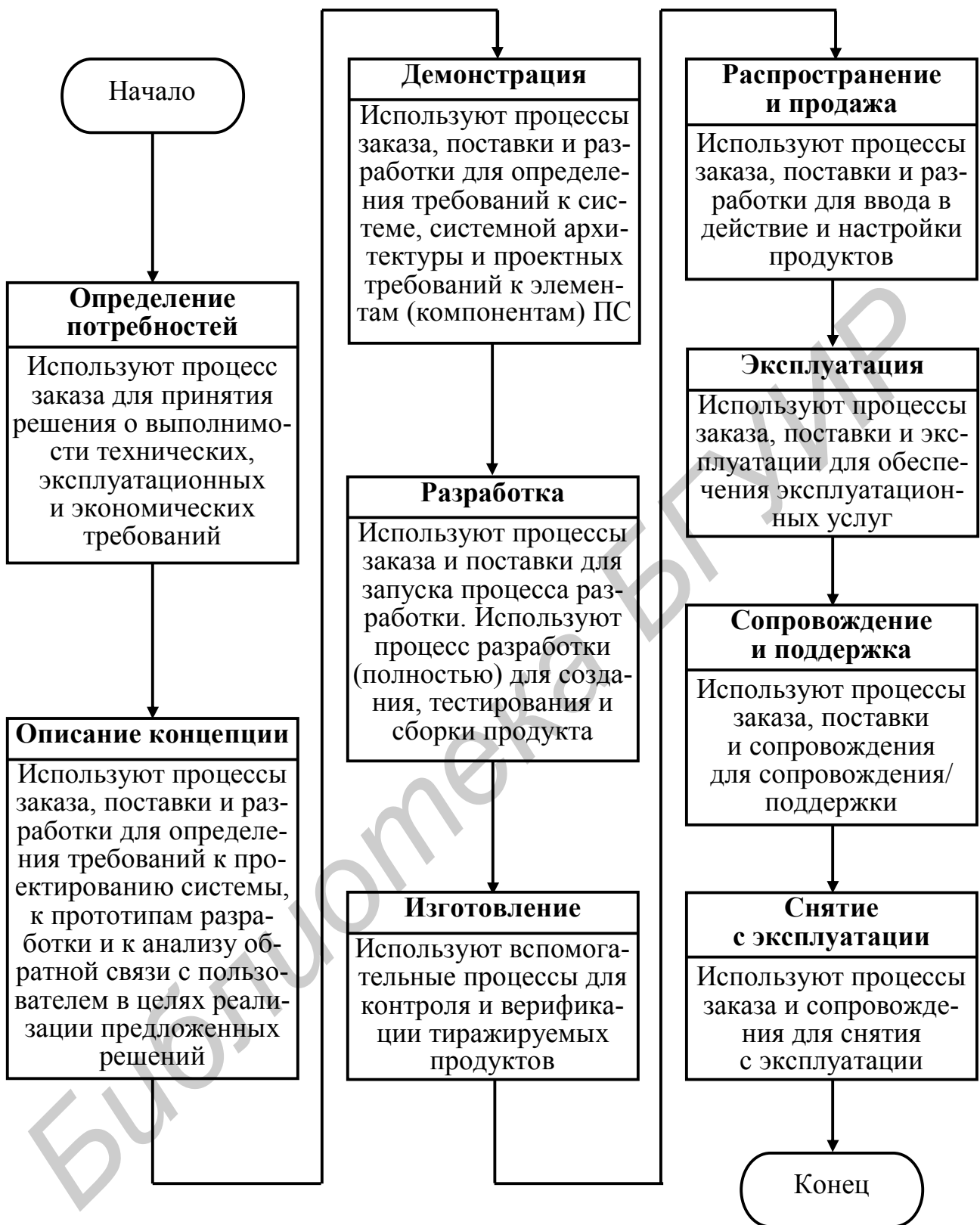


Рис. 36. Использование ИСО/МЭК 12207  
для обеспечения модели жизненного цикла системы  
в соответствии с ГОСТ Р ИСО/МЭК ТО 15271–2002

Для ПС работа по изготовлению состоит из копирования (тиражирования) ПС и документов к нему на соответствующие носители для различных пользователей.

*На этапе распространения и продажи* систему поставляют заказчику (пользователю) или покупателям (на рынок). *Период распространения* начинается с поставки первой работоспособной системы заказчику. *Период продажи* начинается с поставки первой партии систем на рынок и заканчивается изъятием с рынка.

*На этапе эксплуатации* система применяется пользователями или потребителями. Данный этап заканчивается снятием системы с эксплуатации.

*На этапе сопровождения и поддержки* выполняется модификация системы в связи с наличием в ней ошибок, дефектов, возникновением проблем или необходимостью в адаптации или усовершенствовании.

*На этапе снятия с эксплуатации* система снимается с обслуживания, выполняется ее архивирование и обеспечение ограниченной поддержки.

## **4.6. Процессы качества и требования к оценке**

В *Приложении А* стандарта *ИСО/МЭК ТО 15271* приведены рекомендации по использованию вспомогательных процессов ЖЦ, связанных с качеством ПС (см. подразд. 2.5 пособия). Такими процессами являются:

- обеспечение качества;
- верификация;
- аттестация;
- совместный анализ;
- аудит.

Данные процессы на рис. 3 (см. подразд. 2.1 пособия) выделены серым фоном.

При реализации основных процессов могут быть привлечены как вышеперечисленные вспомогательные процессы, так и дополнительные задачи по оценке, предназначенные для повышения качества выполнения других задач, работ и процессов.

Требования ко всем вышеперечисленным видам оценок приведены в *Приложении В* стандарта *ИСО/МЭК 12207* и пояснены в подразд. 2.5 пособия.

При выполнении небольших проектов применение всех этих возможностей повышения качества может оказаться излишним. При реализации крупных критичных проектов необходимо применение всех соответствующих оценок. С учетом этого перед выполнением проекта необходимо адаптировать процессы, связанные с качеством. С этой целью подготавливается *план обеспечения качества* (см. задачи 6.3.1.1 – 6.3.1.3 в табл. 20 пособия) и при необходимости в виде отдельных документов планы верификации и аттестации (см. задачу 6.4.1.5 в

табл. 21 и задачу 6.5.1.4 в табл. 22 пособия).

В малых проектах план обеспечения качества должен допускать проведение верификации и аттестации группой проектантов и предусматривать элементарный процесс совместного анализа.

В больших критичных проектах должны быть запланированы независимые группы по верификации и аттестации, а также совместные анализы и аудиторские проверки.

## 4.7. Модели жизненного цикла

В Приложении С стандарта ИСО/МЭК ТО 15271 описаны *три фундаментальных модели ЖЦ* (их принято называть также *базовыми стратегиями разработки*):

- каскадная;
- инкрементная;
- эволюционная.

Каждая из них может использоваться самостоятельно или комбинироваться с другими для создания *гибридной* модели ЖЦ.

Три базовых стратегии могут быть реализованы различными моделями ЖЦ.

### 4.7.1. Каскадная модель

Каскадная фундаментальная модель ЖЦ системы реализует *принцип однократного выполнения* следующих видов деятельности:

- установление потребностей пользователя;
- определение требований;
- проектирование системы;
- изготовление системы;
- испытание;
- корректировка;
- поставка или использование.

Таким образом, каскадная модель основана на *полном формулировании требований к системе в начале процесса разработки*. При применении каскадной стратегии работы и задачи процесса разработки выполняются последовательно. Они могут выполняться также частично параллельно в случаях перекрытия последовательных работ.

При одновременной разработке нескольких программных объектов работы и задачи процесса разработки могут выполняться параллельно.

Процессы сопровождения и эксплуатации реализуются после разработки. Процессы заказа, поставки, вспомогательные и организационные процессы выполняются параллельно с процессом разработки.

На рис. 37 приведен пример каскадной модели ЖЦ системы [8].

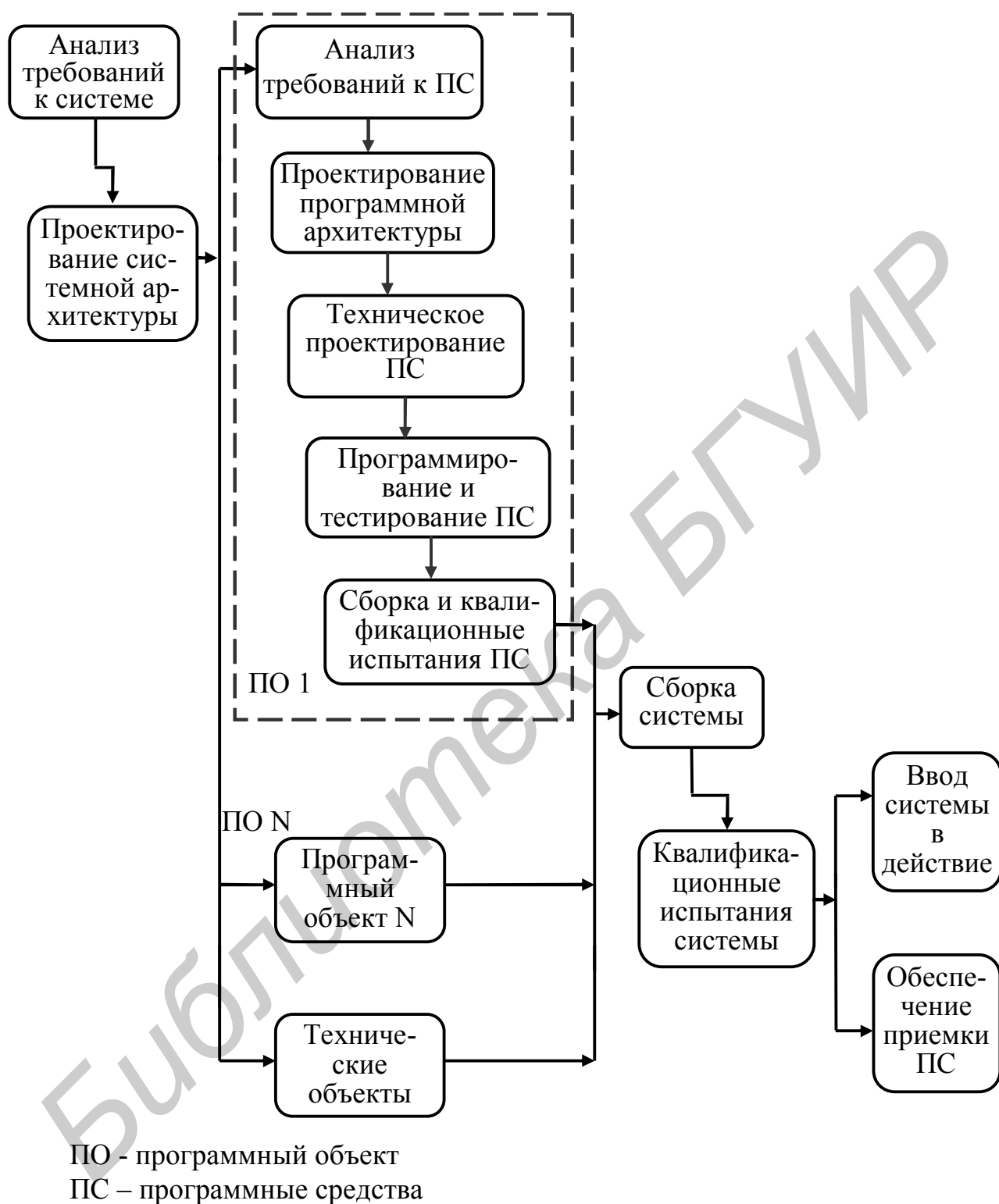


Рис. 37. Вариант каскадной модели  
 по ГОСТ Р ИСО/МЭК ТО 15271–2002

Данный вариант учитывает возможности параллельной разработки программных объектов, входящих в состав системы.

*Достоинства каскадной модели:*

- однократное представление всех характеристик системы (однократное представление требований и продукта разработки);
- необходимость единственной фазы перехода от старой системы к новой (один проход процесса разработки для перехода к новой системе).

*Использование каскадной модели затруднено в следующих случаях:*

- требования к объектам определены недостаточно четко;
- большой объем системы; это делает практически невозможным однократное выполнение работ по её созданию;
- предполагаются скорые изменения в технологии работ; это повышает риски нарушения графика проекта и разработки системы, не удовлетворяющей требованиям;
- возможны текущие изменения требований к системе; это вызывает необходимость разработки системы заново;
- ресурсы, например средства или персонал, ограничены;
- необходимость использования промежуточных продуктов разработки; последние при применении каскадной модели для использования непригодны.

С вышеперечисленными ограничениями связаны основные недостатки каскадной модели.

#### **4.7.2. Инкрементная модель**

Инкрементная фундаментальная модель называется также *запланированным усовершенствованием продукта*.

При ее использовании *набор требований к системе формулируется полностью в начале процесса разработки*. Затем реализуется разработка последовательности инкрементов (конструкций). Первый инкремент реализует часть требований. В последующих инкрементах *в соответствии с планом требования реализуются постепенно с небольшими добавлениями* по отношению к предыдущим инкрементам. Процесс продолжается, пока не будет закончено создание системы.

При использовании данной фундаментальной модели некоторые процессы, работы и задачи могут выполняться однократно, а другие – при создании каждого инкремента. При разработке каждого инкремента, как правило, используется каскадная модель с возможностью частичного перекрытия работ.

Процессы сопровождения и эксплуатации могут реализовываться параллельно с процессом разработки. Процессы заказа, поставки, вспомогательные и организационные процессы выполняются параллельно с процессом разработки.

На рис. 38 приведен вариант реализации инкрементной фундаментальной модели [8]. В данном варианте модели предусмотрена возможность как последовательного выполнения работ по разработке инкрементов (например, разработка инкремента 2 и инкремента N на рис. 38), так и частично параллельного

выполнения с перекрытием (разработка инкрементов 1 и 2).

В рассматриваемом варианте инкрементной модели однократно выполняется только работа по анализу требований к системе (см. работу 5.3.2 на рис. 7 пособия). Данная работа и ее результаты на рис. 38 обозначены буквой Т. Все остальные работы процесса разработки стандарта *ИСО/МЭК 12207* повторяются при разработке инкрементов.

На этапе проектирования выполняется проектирование системной архитектуры инкремента, анализ требований к программным средствам, проектирование программной архитектуры, техническое проектирование программных средств инкремента (см. работы 5.3.3 – 5.3.6 на рис. 7 пособия).

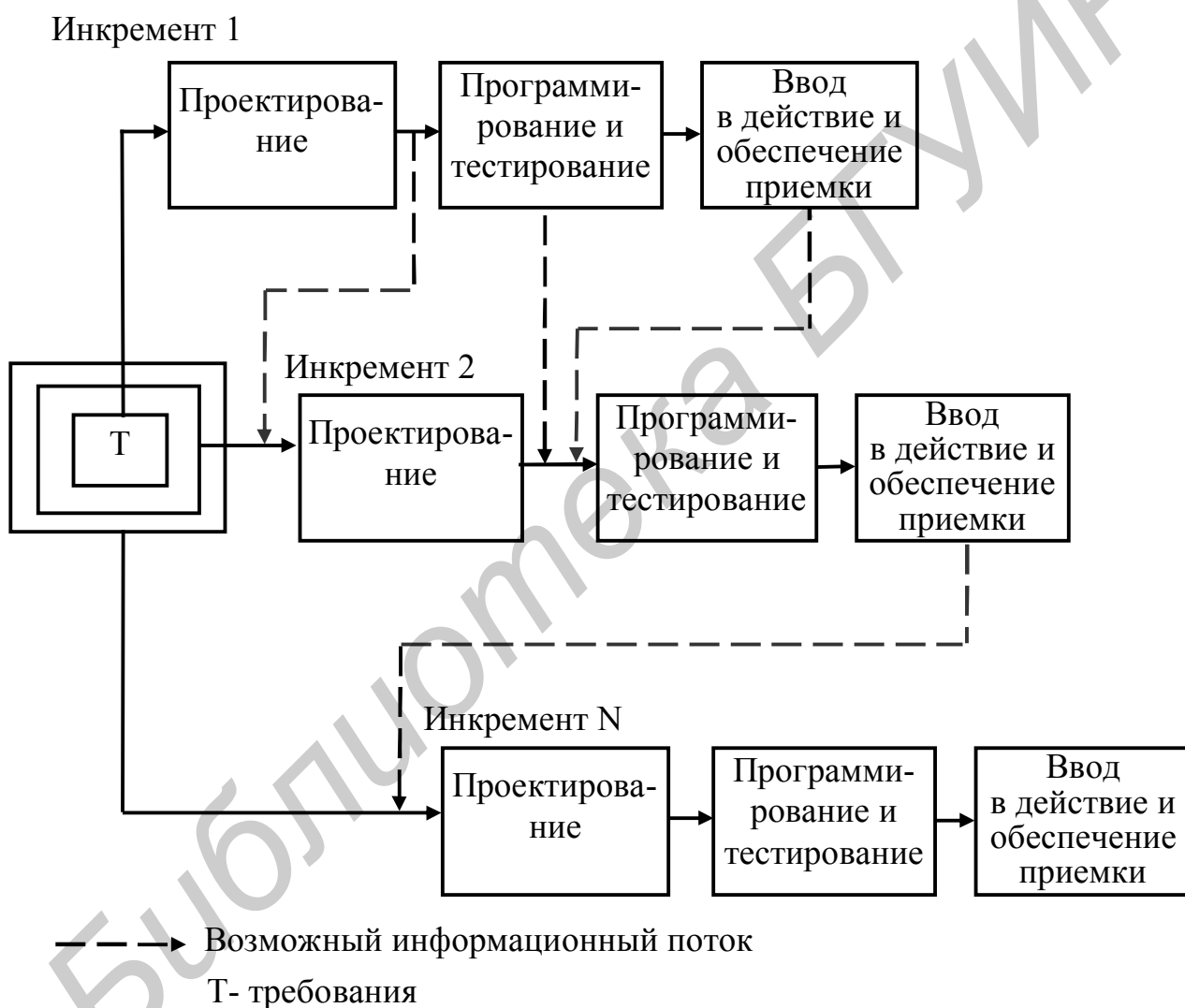


Рис. 38. Вариант инкрементной модели  
по ГОСТ Р ИСО/МЭК ТО 15271–2002

На этапе программирования и тестирования выполняется программирование, тестирование, сборка и квалификационные испытания программных средств, сборка и квалификационные испытания инкремента (соответствует ра-

ботам 5.3.7 – 5.3.11 на рис. 7).

На этапе ввода в действие и обеспечения приемки выполняется ввод инкремента в действие в среде эксплуатации и обеспечение его приемки пользователем (соответствует работам 5.3.12 – 5.3.13 на рис. 7).

*Достоинства инкрементной модели:*

- необходимость изначального формулирования характеристик системы (требований к ней);
- пригодность для использования промежуточного продукта;
- естественное разделение системы на наращиваемые компоненты (инкременты);
- возможности наращивания привлекаемого персонала и средств.

*Использование инкрементной модели затруднено в следующих случаях:*

- требования к объектам определены недостаточно точно;
- необходимо предусмотреть сразу все возможности системы (реализовать все требования к ней);
- возможны текущие изменения требований к системе;
- возможности привлечения ресурсов (средств или персонала) на длительный период ограничены;
- предполагаются скорые изменения в технологиях работ.

С вышеперечисленными ограничениями связаны основные недостатки инкрементной модели.

### **4.7.3. Эволюционная модель**

При использовании эволюционной фундаментальной модели ЖЦ, как и в предыдущем случае, система разрабатывается в виде последовательности конструкций (версий). Однако *требования изначально определяются только частично и уточняются в каждой последующей версии.*

При разработке каждой версии, как правило, используется каскадная модель ЖЦ с возможностью частичного перекрытия работ или задач.

Работы и задачи процесса разработки обычно выполняются многократно. Процессы сопровождения и эксплуатации могут быть реализованы параллельно с процессом разработки. Процессы заказа, поставки, вспомогательные и организационные процессы выполняются параллельно с процессом разработки.

Один из вариантов эволюционной модели приведен на рис. 39. На данном рисунке используются те же этапы процесса разработки, что и на рис. 38.

*Достоинства эволюционной модели:*

- итерационное определение возможностей системы (требований к ней);
- пригодность для использования промежуточного продукта;
- естественное разделение системы на наращиваемые компоненты;
- привлечение персонала и средств по мере необходимости;
- необходимая обратная связь с пользователем для полного понимания требований;

- упрощение надзора за изменением технологии.

*Использование эволюционной модели затруднено в следующих случаях:*

- все возможности системы (требования к ней) определены изначально;
- возможности долговременного привлечения ресурсов (например средств или персонала) ограничены.

С вышеперечисленными ограничениями связаны основные недостатки эволюционной модели.

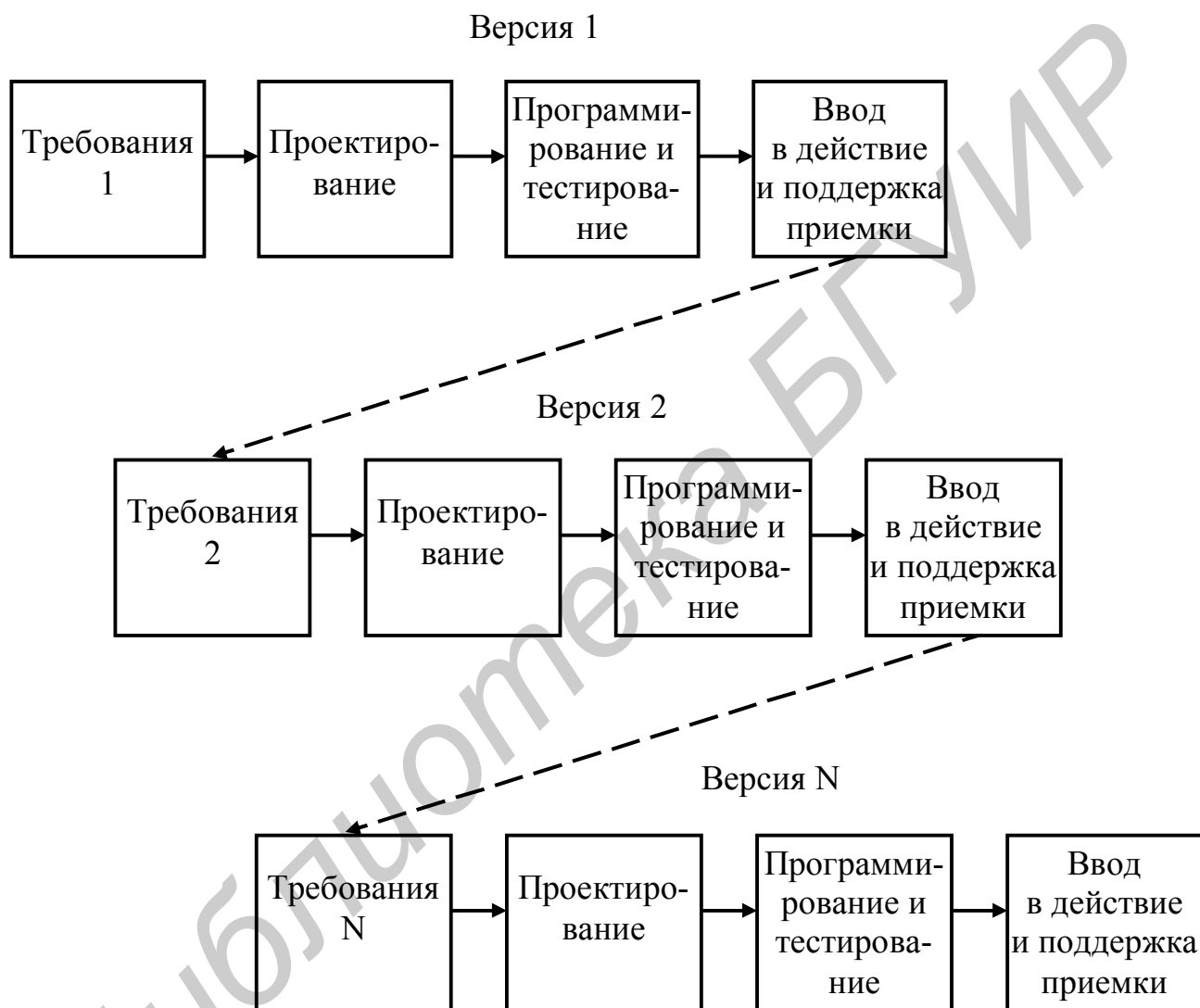


Рис. 39. Вариант эволюционной модели по ГОСТ Р ИСО/МЭК ТО 15271–2002

## 4.8. Примеры адаптации ИСО/МЭК 12207

Приложение D стандарта ИСО/МЭК ТО 15271 содержит несколько примеров адаптации требований стандарта ИСО/МЭК 12207 к условиям конкретного применения. Данные примеры основаны на использовании макетирования. В



настоящее время макетирование достаточно часто применяется в современных реализациях моделей ЖЦ.

#### 4.8.1. Пример макетирования небольшой системы

При разработке небольших систем полное применение стандарта *ИСО/МЭК 12207* может оказаться излишним, поскольку потребует нецелесообразных затрат времени, стоимости и других ресурсов. В этом случае наиболее экономически эффективным решением при выборе модели ЖЦ может оказаться макетирование (прототипирование) системы.

На рис. 40 приведен пример адаптации стандарта *ИСО/МЭК 12207* к модели ЖЦ, основанной на макетировании небольшой системы. На данном рисунке в скобках приведены номера работ или процессов вышеназванного стандарта, входящих в состав соответствующих стадий модели ЖЦ.

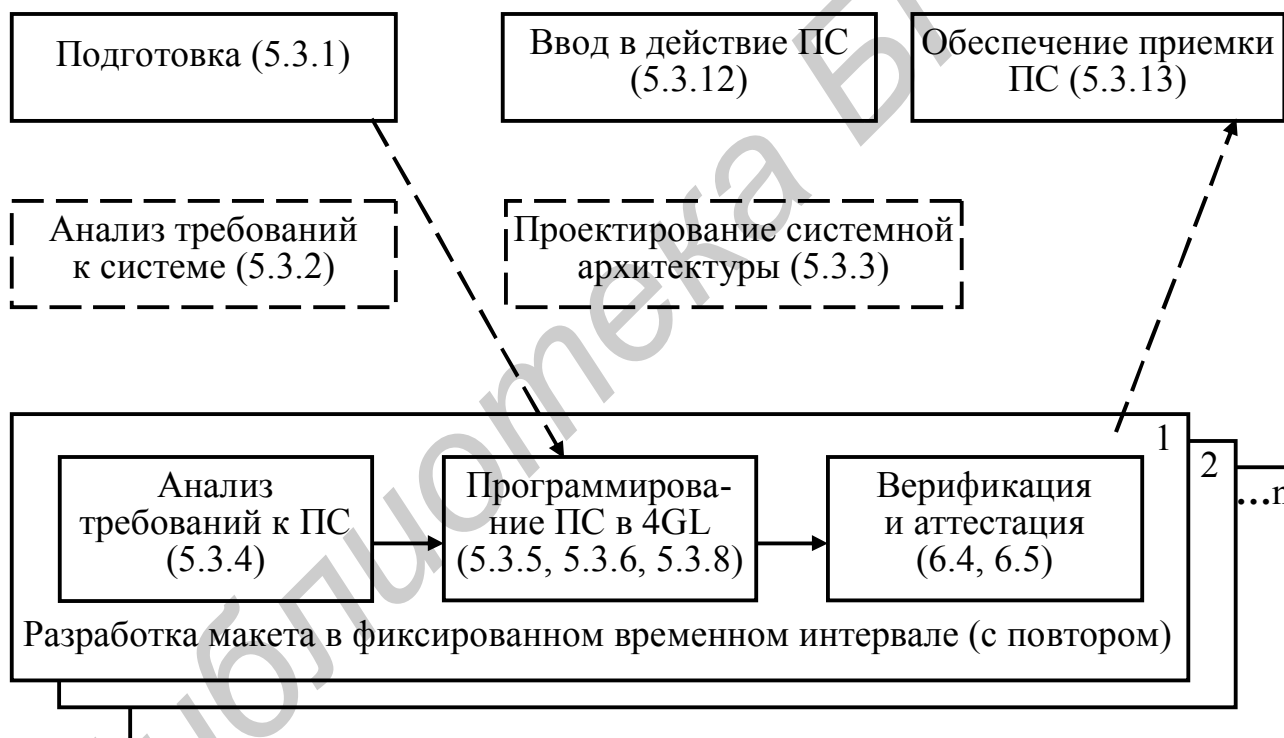


Рис. 40. Пример адаптированной модели ЖЦ, использующей макетирование, по ГОСТ Р ИСО/МЭК ТО 15271–2002

Основой эффективного макетирования системы является максимально возможная детализация на ранних стадиях проекта (анализ требований к системе и проектирование системной архитектуры). Данные стадии выполняются в модели ЖЦ один раз и только в том случае, если ПС входит в состав системы. Однократное выполнение этих стадий достигается за счет тесных связей разработчиков с пользователями проекта. Требования к системе, в первую очередь,

функции системы и внешние интерфейсы определяются пользователями в начале ЖЦ, деловые процессы уточняются при проведении пользователем серии оценок прототипов системы.

Рассматриваемая модель ЖЦ основана на применении эволюционной стратегии разработки ПС, в которой при создании каждой версии ПС вместо каскадной модели используется прототипирование. При разработке каждого прототипа уточняются требования к нему. Для быстрого проектирования и сборки ПС (работы 5.3.5, 5.3.6, 5.3.8 процесса разработки ИСО/МЭК 12207, см. рис. 7, пособия), а также оперативного наращивания, изменения и уточнения ПС используется инструментальная система 4GL (Fourth Generation Language – язык программирования четвертого поколения). Языки 4GL осуществляют автоматическую кодогенерацию ПС, поэтому работа 5.3.7 процесса разработки (программирование и тестирование ПС, см. рис. 7) в модели ЖЦ не учитывается.

Проверка и оценка каждого макета (прототипа) осуществляется пользователем в реальной эксплуатационной среде.

В модели определен фиксированный период проведения макетирования и произвольное количество итераций.

Разработчик ПС *контролирует макетирование* с помощью:

- установления приоритетов требований к ПС;
- ужесточения ограничений временного интервала;
- привлечения конечного пользователя.

#### **4.8.2. Пример ускоренной разработки приложения**

В данном примере макетирование (прототипирование) применяется для полной разработки большой системы. Для ускорения разработки используется RAD-модель ЖЦ (RAD – Rapid Application Development – ускоренная разработка приложения).

Для успешной реализации RAD-модели разработчики должны тесно взаимодействовать с конечными пользователями, иметь навыки работы с соответствующими технологиями и инструментальными средствами, а область применения приложения не должна быть критичной.

*Основные факторы успешной реализации RAD-модели:*

- установление приоритетов требований, определяющих качество эксплуатационных характеристик системы;
- анализ создаваемых продуктов, ориентированный на виды осуществляемой деятельности;
- использование строгих процедур управления конфигурацией;
- обоснование состава персонала, необходимого для достижения более широких целей, чем сформулированные задачи;
- проведение испытаний на всем протяжении ЖЦ объекта;

- обоснование временных и стоимостных оценок функциональных возможностей конечных продуктов, более широкое, чем установленный состав работ по их созданию;
- углубленная оценка риска при функционировании системы, более детальная по сравнению с анализом структуры системы;
- установление общих требований в виде, пригодном для декомпозиции во время разработки.

На рис. 41 приведен один из вариантов RAD-модели ЖЦ ПС, адаптированный под требования стандарта *ИСО/МЭК 12207* [8]. На данном рисунке в скобках обозначены номера процессов, работ или задач, принятые в стандарте *ИСО/МЭК 12207* (см. пп. 2.2.3, 2.2.4 пособия) и используемые на соответствующих этапах модели.

*На этапе осуществимости* выполняется анализ проекта на основе критических факторов успешности реализации RAD-модели.

*На этапе анализа деловой деятельности* определяется область применения проекта и разрабатывается план макетирования (прототипирования).

*В цикле функциональной модели* с помощью специальных инструментальных средств быстрой разработки (CASE-средств) разрабатывается функциональный прототип (функциональная модель) и формулируются нефункциональные требования и стратегия реализации прототипа.

*В цикле проектирования и создания* создается система, прошедшая квалификационные испытания и удовлетворяющая всем функциональным и нефункциональным требованиям.

*На этапе реализации* выполняется внедрение системы в среде пользователя, ее документирование и обучение соответствующего персонала.

Цикл функциональной модели и цикл проектирования и создания предусматривают последовательную итерацию работ 5.3.3 – 5.3.9 стандарта *ИСО/МЭК 12207* (см. рис. 7). При этом каждая итерация прототипа должна быть создана в соответствии с графиком работ за определенный временной интервал. Конкретное время реализации каждого шага итерационного цикла, как правило, определяется набором *трех итераций* – предварительное исследование, уточнение и утверждение (принятие).

Из описания приведенного варианта RAD-модели следует, что он фактически реализует эволюционную стратегию разработки. В каждом цикле стратегии реализуется уточнение функциональных и нефункциональных требований, разработка функционального прототипа, разработка версии системы, ввод в действие и эксплуатация данной версии.

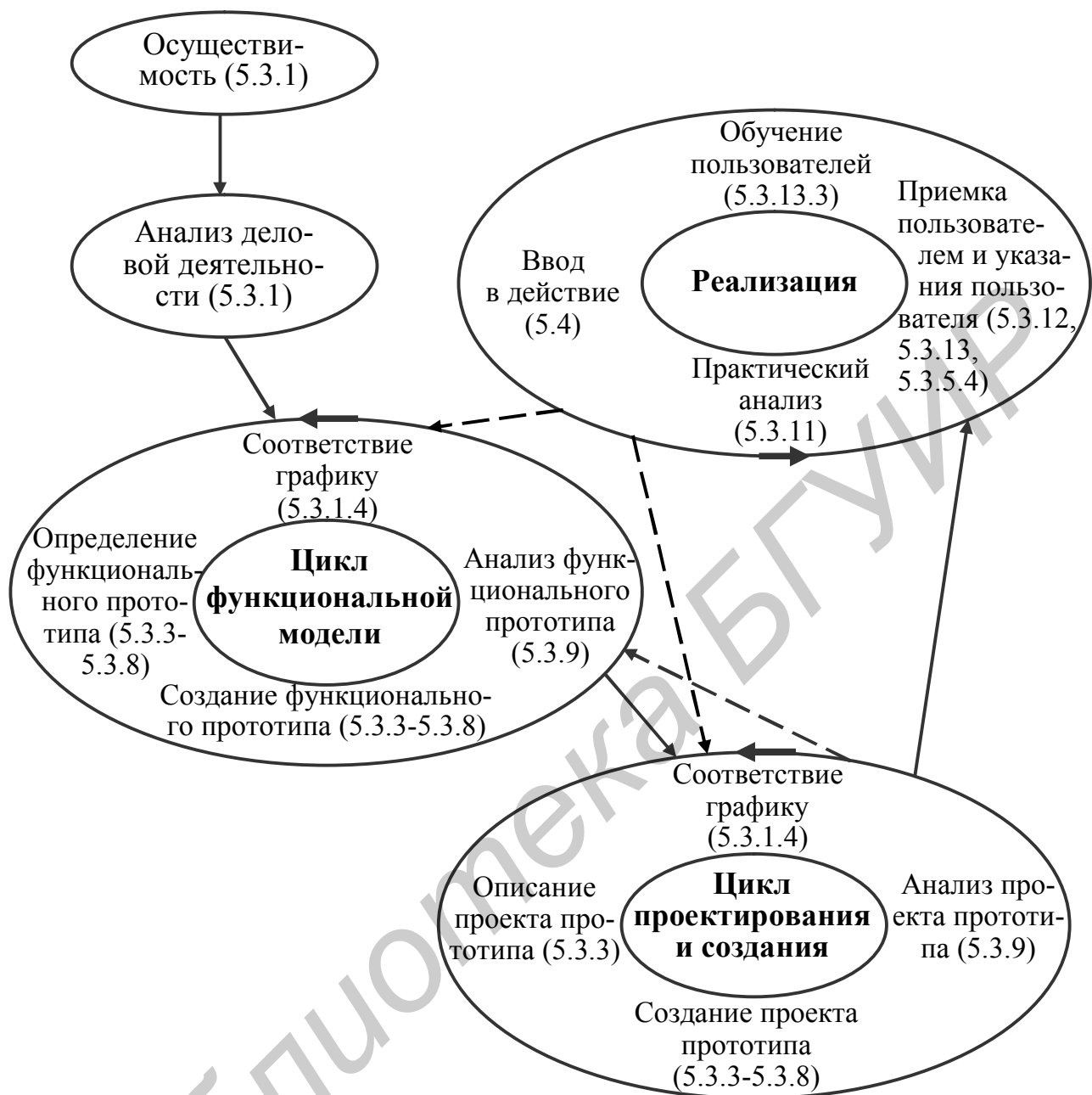


Рис. 41. Вариант RAD-модели быстрой разработки приложений по ГОСТ Р ИСО/МЭК TO 15271-2002

## 5. СТАНДАРТ ISO/IEC TR 16326

### 5.1. Общие сведения

Международный стандарт *ISO/IEC TR 16326:1999 – Программная инженерия – Руководство по применению ISO/IEC 12207 при управлении проектом* [43] – введен в действие в 1999 г. В России аутентичный стандарт *ГОСТ Р ИСО/МЭК ТО 16326–2002* действует с 2003 г. [9].

В данных стандартах приведены уточнения и дополнения к стандарту *ИСО/МЭК 12207* в части процесса управления, а также рекомендации по его использованию. Стандарт *ИСО/МЭК ТО 16326* предназначен в первую очередь для администраторов проектов, отвечающих за управление реализацией основных процессов ЖЦ ПС, определенных в *ИСО/МЭК 12207*: заказа, поставки, разработки, эксплуатации и сопровождения.

Стандарт *ИСО/МЭК ТО 16326* разработан на основе:

- процесса управления, определенного стандартом *ИСО/МЭК 12207*;
- Руководства *PMBOK* [53];
- стандарта *ISO 10006–1997* [16].

В Руководстве *PMBOK* содержится информация по управлению проектами в целом. В стандарте *ИСО/МЭК 12207* приведены информация о программных проектах в целом, процессах жизненного цикла программных средств и описание подлежащих реализации работ и задач. В стандарте *ISO 10006* представлена информация, относящаяся к повышению качества управления проектом. Три вышеуказанных документа в стандарте *ИСО/МЭК ТО 16326* взаимодополняют друг друга.

Стандарт *ИСО/МЭК ТО 16326* имеет следующую структуру:

1. Область применения.
2. Соответствие.
3. Нормативные ссылки.
4. Определения.
5. Обозначения.
6. Руководство.

Приложение А (справочное). Обеспечение процесса управления по *ИСО/МЭК 12207*.

Приложение В (справочное). Работы управления программным проектом, связанные с работами процесса управления.

Приложение С (справочное). Процессы управления проектом, соответствующие работам процесса управления по *ИСО/МЭК 12207*.

Приложение D (справочное). Дополнительная информация.

Приложение E (справочное). Библиография.

*Приложение А* содержит таблицу, отражающую, как основные процессы, работы и задачи ЖЦ ПС из стандарта *ИСО/МЭК 12207* обеспечивают (поддерживают) работы процесса управления.

*В Приложении В* содержится таблица, определяющая процессы из Руководства *РМВОК*, применяемые при работах процесса управления из стандарта *ИСО/МЭК 12207*.

*Приложение С* содержит таблицу, определяющую процессы управления проектом из стандарта *ИСО 10006*, применяемые при работах процесса управления из стандарта *ИСО/МЭК 12207*.

*Приложение D* содержит сводную таблицу, в которой отражается взаимосвязь работ ЖЦ из стандарта *ИСО/МЭК 12207*, процессов из *ИСО 10006* и областей сведений из Руководства *РМВОК*. Приложение содержит также определение основных терминов из стандарта *ИСО 10006* и Руководства *РМВОК*, которые используются в стандарте *ИСО/МЭК ТО 16326*.

В данном разделе учебного пособия приведены основные положения стандарта *ИСО/МЭК ТО 16326*.

## **5.2. Основные определения**

В этом подразделе рассмотрены основные определения, приведенные в стандартах *ИСО/МЭК ТО 16326*, *ИСО 10006* и Руководстве *РМВОК* [9, 16, 53].

**Проект** – деятельность по созданию продукта, услуги и выходных результатов, характеризующаяся следующими свойствами:

- в проект вовлекаются группа лиц, другие ресурсы и мероприятия;
- проект имеет начало и конец, то есть носит временной характер;
- проект не связан с обычной деятельностью организации, то есть носит индивидуальный характер.

Примерами ресурсов являются программные средства, оборудование, прочие средства, финансы, информационные системы, материалы, персонал, услуги, площади.

**Отличия программных проектов** от других проектов:

- программные средства являются наиболее сложными;
- внести изменения в программное средство относительно просто;
- большинство обнаруженных проблем с техническими средствами решают путем изменения программных средств;
- в связи с низкой стоимостью тиражирования для программных средств отсутствует установленный технологический процесс;
- программирование не связано с традиционными естественными науками, поэтому аналогичные методы тестирования и проектного моделирования отсутствуют;
- программные средства являются элементами системы, увеличивающими ее сложность и создающими предпосылки для ее последующих изменений;

- программные средства наиболее доступны для пользователей и поэтому являются основным объектом их претензий.

В процессе управления проектом по *ИСО/МЭК 12207* используются следующие **виды управления**, определенные в Руководстве *РМВОК*:

- *управление обменом в проекте* – охватывает деятельность, необходимую для сбора и распространения проектной информации;
- *управление стоимостью проекта* – охватывает деятельность по обеспечению организации проекта в рамках установленного бюджета;
- *управление людскими ресурсами проекта* – охватывает деятельность по эффективному использованию персонала, вовлеченного в проект;
- *управление интеграцией (сборкой) проекта* – охватывает деятельность по соответствующей координации между различными элементами проекта;
- *управление организацией проекта* – охватывает деятельность по заказу продуктов и услуг в организации, их предоставляющей;
- *управление качеством проекта* – охватывает деятельность, обеспечивающую соответствие проекта установленным потребностям; включает в себя планирование качества, обеспечение качества и контроль качества;
- *управление риском проекта* – охватывает деятельность, связанную с определением, анализом проектного риска и ответственностью за него; включает в себя определение риска, его количественную оценку, разработку ответных действий на риск и контроль этих действий;
- *управление областью проекта* – охватывает деятельность по включению в проект задач жизненного цикла (по *ИСО/МЭК 12207*), необходимых и достаточных для успешной реализации проекта;
- *управление сроками проекта* – охватывает деятельность по обеспечению своевременного завершения проекта.

В вышеприведенном списке используется термин «деятельность». Данному термину в Руководстве *РМВОК* и стандарте *ИСО 10006* соответствует понятие «процессы». Однако это понятие не совпадает с понятием процесса в стандарте *ИСО/МЭК 12207*. Для того чтобы избежать путаницы в терминологии, в этом подразделе применен аналогичный в данном контексте термин «деятельность».

В соответствии с *ИСО 10006* термин **область проекта** охватывает описание проектных продуктов, их характеристик и способов их измерения или оценки. Деятельность, связанная с областью проекта, предназначена для:

- переноса требований заказчика в работы, выполняемые для достижения целей проекта, и для организации самих работ;
- обеспечения персонала работой в заданной области при реализации конкретных работ;
- обеспечения соответствия работ требованиям, установленным в области применения проекта.

С областью проекта связаны следующие **виды деятельности**:

- *разработка концепции* – определение основополагающих принципов, по которым должны быть созданы проектные продукты;

- *определение области проекта и ее контроль* – документирование характеристик проектных продуктов в терминах их измеримости и контролируемости;
- *определение работ* – установление и документирование работ и этапов, необходимых для достижения целей проекта;
- *контроль работ* – контроль фактической работы, выполняемой в проекте.

### 5.3. Руководство по процессу управления

В стандарте *ИСО/МЭК 12207* определен общий процесс управления, более широкий, чем управление программным проектом. В рассматриваемом стандарте *ИСО/МЭК ТО 16326* анализируется применение процесса управления из *ИСО/МЭК 12207* непосредственно для управления программным проектом.

Ответственным за выполнение процесса управления является администратор проекта.

Первой работой процесса управления в соответствии с *ИСО/МЭК 12207* является ***подготовка и определение области управления*** (см. работу 7.1.1 в табл. 26 пособия). Данная работа должна заключаться в следующем:

- установление выполнимости процесса при наличии, достаточности и соответствии персонала, материалов, средств, среды программной инженерии и технологий, необходимых для выполнения проекта и управления им;
- определение обоснованных (экономически, повременно и технически) сроков реализации проекта.

При этом производится выбор *сценария разработки проекта*. Например, проект состоит из:

- готовых ПС;
- ПС собственной разработки;
- ПС посторонней разработки;
- комбинации предыдущих вариантов.

*Область управления*, охватывающая описание проектных продуктов, их характеристик и способов их измерения или оценки, связана со следующими вопросами:

- документирование обоснования проекта, его целей и задач;
- перевод требований к проекту в результаты проекта и работы ЖЦ;
- обеспечение персонала заданиями;
- оценка результатов работ.

Должны быть предусмотрены специальные мероприятия по определению и документированию *характеристик качества результатов проекта*. При этом должны быть учтены требования и рекомендации серии базовых международных стандартов *ISO/IEC 9126-1-4* в области оценки качества ПС [49 – 52].

Должны быть определены обязанности по согласованию с заказчиком тре-



бований к проекту. При выполнении сложных проектов данное согласование достигается в результате итерационного выполнения процессов, работ и задач в ЖЦ проекта. Итерации необходимы, чтобы уменьшить следующие *виды рисков*:

- изменения в требованиях заказчика;
- изменения в среде проектирования;
- изменения в бюджете и графике работ по проекту;
- изменения в процессе проектирования.

При выполнении итераций важно установить критерии завершения проекта.

Должны быть определены деловые требования, регламентирующие авторские права, вопросы патентования и т.д.

*Советы по специфике программных средств при подготовке и определении области управления:*

- должны быть определены требования по тиражированию, распространению, вводу в действие (инсталляции) и тестированию ПС;
- должна быть установлена и отслежена взаимосвязь (трассировка) между требованиями к системе и требованиями к ПС, требованиями к ПС и проекту, требованиями к ПС и тестированию;
- должны быть установлены, проконтролированы и задокументированы соответствующие интерфейсы как неотъемлемая часть спецификации требований к ПС;
- должна быть учтена сложность подтверждения соответствия ПС установленным и предполагаемым требованиям пользователя вследствие сложности разработки ПС;
- проектирование рабочей загрузки при выполнении работ проекта должно зависеть от *типа проекта*:
  - новая разработка;
  - встраивание или интеграция в систему;
  - модификация готового ПС;
  - подключение к различным операционным системам.

Второй работой процесса управления в соответствии с *ИСО/МЭК 12207* является **планирование** (см. работу 7.1.2 в табл. 26 пособия). Минимальное содержание планов выполнения процессов, определенное вышеназванным стандартом, приведено в п. 2.4.1 пособия.

Планы должны устанавливать также модель ЖЦ ПС, задачи, распределение задач, их блокировку (обоснование отсутствия необходимости в их выполнении) и соответствующие ресурсы. В проекте должен быть установлен один основной график работ, все вспомогательные графики должны быть согласованы с ним. Должно быть обеспечено управление программным проектом на различных уровнях его детализации с использованием соответствующих технологий в зависимости от объема, сложности, критичности и риска проекта.

*Оценки проекта*, используемые при планировании, должны охватывать:

- стоимость реализации соответствующих процессов;
- инфраструктуру;
- потребности в ресурсах;
- оценку и контроль качества;
- управление риском;
- обеспечение среды программной инженерии;
- задания, выполняемые в каждом процессе или работе.

Планирование является постоянной, повторяющейся работой, при выполнении которой оценивают, уточняют и корректируют ход проекта.

Таким образом, дополнительно к минимальному содержанию, определенному в стандарте *ИСО/МЭК 12207*, планы выполнения процессов должны содержать:

- роли и обязанности соответствующих сторон, участвующих в проекте;
- выбор модели ЖЦ ПС;
- подлежащие выполнению работы и задачи;
- перечни всех проектных результатов (продуктов), подлежащих поставке;
- критерии завершения работ и задач;
- перечни окончательных отчетных материалов;
- перечень отчетных материалов по стоимости и графикам проведения работ;
- средства организации работ по управлению, выпуску продукта и синхронизации работ;
- периодичность и способы выдачи отчетных материалов;
- перечень отчетных материалов по проблемам и выполнению работ и задач;
- требования к ресурсам и их наличие.

Соответствующие планы должны быть разработаны администраторами процессов и привязаны к плану управления программным проектом (см. задачу 5.2.4.5 в табл. 2 и содержание плана управления в п. 2.2.2 пособия). Они могут быть оформлены в виде отдельных планов или включены в план управления программным проектом. Администратором проекта ПС должна быть определена отчетность по вспомогательным процессам.

При планировании критериев управления качеством ПС (см. содержание планов выполнения процессов, приведенное в п. 2.4.1 пособия) определяются *контрольные точки проекта* и результаты, установленные для этих точек. Для данных точек должны быть запланированы необходимые отчеты и выходные результаты вспомогательных процессов.

Основным источником для анализа возможностей и эффективности процессов в организации, рабочих характеристик данного проекта и обобщения полученного опыта являются *архивные данные по проекту*. Эти архивные данные должны образовывать общую базу данных организации. Их следует посто-

янно использовать для усовершенствования процессов ЖЦ этой организации и отдельных проектов. В каждой организации должна быть создана официальная система для сбора, анализа, обобщения, архивирования и поиска архивных данных по проектам.

*Советы по специфике программных средств при планировании:*

- в проектах программных средств стратегия эффективного управления конфигурацией является ключевой (критичной); поэтому для управления изменениями ПС могут быть использованы следующие стратегии:
  - определение пороговой величины изменения в стоимостном выражении, позволяющей вносить изменение в ПС без пересмотра конкретного договора;
  - использование пакетов изменений, облегчающее их внесение в ПС с минимальной корректировкой графика работ и обеспечивающее максимальную стабильность ПС;
  - установление соглашений по взаимосвязям для всего проекта в части проблем, связанных с неясностью, неточностью, изменчивостью или непроверяемостью требований;
- следует учитывать трудности прогнозирования последствий внесения изменений в программные системы и управления этими последствиями;
- большинство моделей стоимости основано на оценке объема ПС; поэтому важной является проблема определения конкретного объема ПС и адаптации (уточнения) заданной модели стоимости к конкретному проекту; при этом должно быть учтено следующее:
  - данные для модели собирают по ранее реализованным проектам;
  - для реализации и интерпретации модели используют экспертов;
  - при отсутствии адаптации модели с учетом опыта организации данная модель может давать ошибку на порядок;
  - при оценке окончательной стоимости проекта не следует использовать только модель стоимости; необходимо учитывать также опыт предыдущих проектов;
- ПС следует представлять в виде пакета; это повышает его эффективность как в целом, так и при тиражировании, распространении, инсталляции, тестировании и эксплуатации;
- планы по ПС должны быть взаимоувязаны с планами по техническим средствам системы и должны контролироваться совместно; должны быть увязаны контрольные точки по данным планам;
- планы по ПС должны быть взаимоувязаны с планами заказчика, а также планами основных технических средств и среды;
- при представлении ПС нужно определить:
  - проводится ли сопровождение данного ПС и допустимы ли его модификации;
  - права собственности (гарантии, интеллектуальные права, патенты, авторские права).

Третьей работой процесса управления в соответствии с *ИСО/МЭК 12207* является **выполнение и контроль** (см. работу 7.1.3 в табл. 26 пособия).

Эта работа начинается после получения администратором программного проекта полномочий по расходованию денежных средств.

При реализации контрольной части данной работы администратор должен отвечать за надзор, обеспечивающий обнаружение и анализ отклонений от запланированного выполнения конкретного контролируемого процесса, и внесение корректировок в ход данного процесса. Инструментарием надзора и контроля выполнения процесса являются *измерения ПС* по соответствующим показателям, контроль вносимого изменения, оценки и аудиторские проверки программных процессов и продуктов. Измерения ПС могут быть использованы для проверки соответствия ожидавшихся функций программного продукта его функциям при эксплуатации, а также для проверки достигнутых уровней характеристик качества и их соответствия требуемым уровням.

В ходе данной работы процесса управления должно выполняться также *управление проектным риском*. Оно охватывает неопределенности при планировании проекта и требует структурированного подхода. Целями данного управления является минимизация воздействия потенциально неблагоприятных событий и выработка предложений по совершенствованию проекта. Риск связан либо с проектными процессами и инструментальными средствами, либо с соответствием продукта целям проекта.

*Советы по специфике программных средств при выполнении и контроле:*

- исправление ошибочного графика работ не должно отрицательно влиять на эффективность, стоимость или риски проекта;
- увеличение персонала по сравнению с предыдущим проектом следует проводить постепенно, в зависимости от возможностей нового персонала;
- для уменьшения риска целесообразно провести демонстрации, позволяющие заказчикам и покупателям оценить функциональные возможности программного продукта до выбора его поставщика (сущность демонстрации описана в подразд. 4.5 пособия);
- целесообразно использовать макетирование для разработки части функций ПС, чтобы продемонстрировать реализуемость его функциональных возможностей в целом (примеры макетирования рассмотрены в подразд. 4.8 пособия);
- работы по проектированию критических систем не следует проводить без наличия достаточных экспертных знаний в предметной области и соответствующей программной инженерии;
- при реализации проекта следует совместно с заказчиком проводить анализы базовой линии требований к ПС, обеспечивающие соответствие целям проекта или их корректировку по стоимости, срокам или эффективности;
- численность персонала и количество рабочих групп должны быть увязаны с объемами финансирования и графиком работ по проекту;
- необходимо установить методы определения эффективности проекта,

позволяющие своевременно обнаружить нарушения установленных ограничений стоимости и графика работ по проекту;

- следует избегать необоснованного изменения персонала при выполнении конкретных заданий;
- необходимы постоянные контакты с заказчиком во избежание неожиданных изменений в стоимости, графике работ и эффективности проекта.

Четвертой работой процесса управления в соответствии с *ИСО/МЭК 12207* является **проверка и оценка** (см. работу 7.1.4 в табл. 26 пособия)).

Работу по проверке и оценке дополняют вспомогательные процессы ЖЦ ПС из стандарта *ИСО/МЭК 12207*.

Администратор программного проекта должен отвечать за *проведение оценок продуктов и планов*:

- по результатам анализа продуктов, работ и задач;
- на соответствие планам, принципам, методологии и технологии управления программным продуктом;
- в части документирования планов и обязательств;
- в части удовлетворения их установленным требованиям;
- в части готовности для перехода к следующему процессу, работе или задаче.

Основой для отслеживания процессов и работ по проекту должны являться планы управления программным проектом.

Необходимо выполнять периодические анализы и оценки хода выполнения и завершения заданий (задач). Администратор должен подтвердить соответствие прогнозируемых и реальных функций программного продукта требуемым функциям. Администратор должен определить набор измерений ПС в ходе ЖЦ ПС. Ход работ должен быть определен по фактическим измерениям объема продукта, объема работ, их стоимости и графику выполнения.

Администратор программного проекта должен разработать также программу измерений ПС для выявления, количественного определения и оценки риска. Необходимо выбрать систему измерений программного риска, обеспечивающую его определение в ЖЦ ПС. Проверки и оценки должны быть проведены для определения технического и финансового риска.

Проверки проектных работ процесса разработки должны быть основаны на проверках функционального или технического уровня (например характеристик качества ПС) и использованы при проведении общей оценки проекта.

Необходимо документально оформить наиболее существенные проблемы, рассмотренные в ходе проверок и оценок, и решения, принятые по ним.

Для управления проектом необходимо выработать рекомендации по элементам, основам и практическому применению *систем качества*.

*Советы по специфике программных средств при проверке и оценке:*

- следует учесть, что следствием любого нарушения графика работ, предшествующих тестированию, без соответствующей корректировки даты поставки может быть недостаточно полное тестирование продукта;

- всеобъемлющий план тестирования должен быть определен в начале ЖЦ проекта;

- должны быть установлены строгие правила регистрации, хранения, модернизации, резервирования и сопровождения программ, тестовых данных и среды тестирования;

- должна быть разработана стратегия возврата к исходному состоянию при тестировании модификаций продукта;

- необходим интегрированный план сборки системы и модулей ПС в соответствии со стратегией выпуска версии системы;

- должно быть представлено подтверждение функциональных возможностей ПС и основной системы, показывающее их явное соответствие потребностям покупателя;

- для обеспечения процесса управления следует ориентироваться на контрольные примеры и их результаты, используемые в процессе разработки.

Пятой работой процесса управления в соответствии с *ИСО/МЭК 12207* является **завершение** (см. работу 7.1.5 в табл. 26 пособия).

В данной работе следует использовать следующие полученные ранее результаты:

- результаты приемочных испытаний, верификации и аттестации ПС и его измерений;

- отчеты об обеспечении качества ПС и результаты эксплуатационного тестирования;

- отчеты о неисправностях ПС;

- результаты аудиторских проверок;

- результаты приемки и окончания процесса, работы или задачи;

- замечания поставщика или заказчика и результаты взаимодействия с ним.

По окончании данной работы должны быть выданы предложения по сопровождению базовых линий и других документов проекта.

Данная работа может выполняться также между процессами, работами и задачами. Например, по окончании этапа проектирования ПС, в который входят работы 5.3.5 – 5.3.6 из стандарта *ИСО/МЭК 12207* (см. рис. 7 пособия), можно провести анализ его результатов и состояния проекта с целью:

- определения возможности перехода к следующему этапу;

- обнаружения и коррекции неправильной стоимости этапа.

При выполнении каждого проекта и по его завершении необходимо обобщить и проанализировать полученный опыт и обеспечить его доступность всей организации для совершенствования соответствующих процессов этой организации. Данные по проекту следует сохранять в базе данных предприятия. Это поможет внедрять методы управления и процессы жизненного цикла ПС при любых изменениях технологий жизненного цикла.

## 6. СТАНДАРТ ISO/IEC 15910

### 6.1. Общие сведения

Возрастающие масштабы применения ПС и их сложность вызывают необходимость в полной, точной и понятной документации на ПС, доступной пользователям. Часто документация разрабатывается после создания соответствующего ПС. Однако с точки зрения ее качества необходимо, чтобы она создавалась в процессе разработки ПС.

С учетом этого в 1999 г. введен в действие международный стандарт *ISO/IEC 15910:1999 – Информационная технология – Процесс создания документации пользователя программного средства* [40]. В России аутентичный стандарт *ГОСТ Р ИСО/МЭК 15910–2002* действует с 2003 г. [7].

Стандарт *ИСО/МЭК 15910* определяет одну из реализаций процесса документирования, описанного в *ИСО/МЭК 12207* (см. п. 2.3.1 пособия), и предоставляет пользователю метод применения данного процесса при создании конкретного ПС. Стандарт содержит структуру комплексного плана разработки документации, однако не определяет состав требований к стилю оформления документов, устанавливая только их диапазоны.

Стандарт *ИСО/МЭК 15910* должен использоваться при создании *пользовательской документации всех видов*:

- печатной документации (руководство пользователя, краткие справочные карты);
- диалоговой (оперативной) информации;
- справочного текста (Help);
- системы диалоговой документации.

Стандарт имеет следующую *структуру*:

1. Область применения.
2. Соответствие.
3. Нормативные ссылки.
4. Определения.
5. Управление качеством (документации).
6. Адаптация.
7. Цели.
8. Требования (к процессу документирования).

Приложение А (справочное). Перекрестные ссылки с *ИСО/МЭК 12207*.

Приложение В (справочное). Использование настоящего стандарта в договоре.

Приложение С (справочное). Образец плана документирования. Документация пользователя для системы АВС – административного управления маг-

нитными лентами.

Приложение D (справочное). Отношения между аудиториями, выданными заданиями, бумажной и диалоговой документацией.

Приложение E (справочное). Рекомендации по написанию на английском языке документации, подлежащей последующему переводу.

Приложение F (справочное). Оценка проекта (документации).

Приложение G (справочное). Оценка плана документирования.

Приложение H (справочное). Образец спецификации стиля.

## 6.2. Основные определения

Стандарт ИСО/МЭК 15910 использует следующие основные определения.

*Аудитория (audience)*: категория пользователей, предъявляющих к документации одинаковые или аналогичные требования, определяющие содержание, структуру и назначение данной документации.

*Документатор (documenter)*: сторона, создающая документацию.

*Справочный текст (help text)*: текст, автоматически выбираемый в зависимости от контекста, в котором он вызывается, облегчающий и увыстряющий при эксплуатации ПС поиск содержащихся в издании объектов. Справочный текст контекстно зависим.

*Диалоговая документация (on-line documentation)*: информация, доступная пользователю при эксплуатации ПС, которая необязательно привязана к конкретному контексту.

*Система диалоговой информации или справочная система (on-line documentation system or help system)*: часть программы или отдельная программа, запрашиваемая пользователем и позволяющая ему просматривать части диалоговой документации или справочного текста.

*Продукт (программный продукт, product)*: полный набор компьютерных программ, процедур и соответствующих им документации и данных, предназначенный для поставки пользователю.

*Тестирование на практичность (usability testing)*: формальный процесс оценки соответствия документации установленным требованиям.

## 6.3. Требования к процессу документирования

Процесс документирования должен выполняться в *два этапа* в соответствии с рис. 42. Работы, выполняемые в пределах этапа, могут производиться параллельно. Работы, выполняемые в различных этапах, должны осуществляться последовательно. Пунктирными линиями показаны возможные итерации работ. В скобках на рисунке указаны исполнители работ.





Рис. 42. Процесс документирования в соответствии со стандартом ГОСТ Р ИСО/МЭК 15910–2002

Из данного рисунка видно, что стандарт *ИСО/МЭК 15910* регламентирует процесс документирования, частично отличающийся от соответствующего процесса, описанного в стандарте *ИСО/МЭК 12207* (см. рис. 14 пособия). Связь задач процесса документирования из стандарта *ИСО/МЭК 12207* (табл. 18 пособия) с пунктами стандарта *ИСО/МЭК 15910* приведена в *Приложении А* последнего.

Для **получения исходных материалов** (первая работа на рис. 42) заказчик должен обеспечивать документатору доступ ко всем материалам проекта, необходимым для подготовки документации, к рабочей копии ПС, аналитикам, программистам и пользователям, а также к копиям применяемых стандартов и руководств по стилю и форматам.

Документатор должен **разработать план документирования** (вторая работа на рис. 42), в котором должны быть определены задания, выполняемые при создании конкретной документации. План документирования должен охватывать комплект документации по всем необходимым видам (см. подразд. 6.1).

В плане документирования должны быть определены область применения, ограничения по использованию, основные решения, касающиеся анализа и проектирования документации, а также процедуры и проверки, выполняемые при разработке документации.

План документирования должен включать определение аудиторий пользователей документации, уровня их образования, способностей, подготовки, опыта и других характеристик, связанных с содержанием, структурой и использованием документации. Каждый тип пользователей должен быть определен отдельно.

При разработке плана документирования следует учитывать *последовательность проектирования документации*, описанную в *Приложении D* стандарта *ИСО/МЭК 15910*. Данная последовательность включает следующие *этапы*:

- анализ пользователей документируемой системы и их объединение в аудитории в соответствии с потребностями пользователей (примеры аудиторий: квалифицированный пользователь, неквалифицированный пользователь, оператор);
- определение заданий, выполняемых каждой аудиторией при работе с ПС; задания удобно представлять в виде иерархической структуры;
- определение перечней потребностей каждой аудитории в информации; перечень потребностей определяется из заданий аудитории; на основе перечней определяется содержание конкретных документов.

После официального согласования плана документатор несет ответственность за его контроль, распространение и последующие изменения. При этом необходимо предусмотреть процедуры аудита всех обновляемых копий плана.

Содержание плана документирования приведено в подразд. 6.4 пособия. Пример плана документирования приведен в *Приложении C* стандарта *ИСО/МЭК 15910*.

**Проверка (оценка) плана документирования** (третья работа на рис. 42) проводится персоналом заказчика, обладающим необходимой квалификацией. При необходимости может быть привлечен документатор.

Целью проверки является гарантирование полноты и правильности представленных материалов и удовлетворения ими потребностей заказчика. Заказчик должен уделять особое внимание структуре, полноте и практичности документации в соответствии с планом-проспектом ее содержания.

План документирования должен быть проверен (оценен) и утвержден до начала работы над первым проектом документации. Рекомендации по его оценке приведены в *Приложении G* стандарта *ИСО/МЭК 15910*. Данное приложение устанавливает *критерии оценки плана документирования заказчиком*. Заказчик

должен установить наличие в плане документирования следующих элементов:

- определение всех аудиторий пользователей;
- содержание (планы-проспекты) документации с оценкой их постраничного объема;
- определение числа печатных копий и методов их печати;
- определение собственника авторских прав на документацию;
- установление методов подготовки документации;
- достаточные сроки для проверки заказчиком редакции документации.

В *Приложении F* стандарта *ИСО/МЭК 15910* представлены рекомендуемые методы оценки проекта документации по прогнозным затратам времени на ее разработку.

*Поминутный и почасовой методы оценки* основаны на временных нормативах на разработку документации. Например, на исследование содержания документации предусматривается 24 часа на проект, разработку плана документирования – 48 часов на проект, проектирование структуры документа – 8 часов на том документации, написание страницы текста – 3 часа, создание каждого графического материала – от 3-х до 5-ти часов, проверку первой редакции документа – 30 минут на страницу, второй редакции – 15 минут на страницу и т.д. С учетом данных нормативов и запланированного объема документов прогнозируется время на их разработку.

*Метод нисходящего проектирования для оценки* затрат времени на разработку документации основан на предположении, что автор может за месяц подготовить 22 страницы нового текста или 44 страницы текста с изменениями. С учетом объема публикации определяется продолжительность ее разработки. Данная продолжительность распределяется на следующие этапы: 15 % – планирование, 50 % – написание первой редакции, 25 % – написание второй редакции, 10 % – подготовка шаблонов (гранок). С учетом продолжительности этапов и состава их работ определяются сроки выполнения каждой работы.

Если прогнозируемые затраты времени на создание документации превышают допустимые, следует привлекать несколько авторов.

Следует отметить, что в общем случае для определения фактических сроков реализации проекта документации недостаточно использовать только результаты методов оценки проекта. Различные методы оценки могут приводить к различным результатам и даже вступать в противоречие друг с другом. Поэтому затраты времени на разработку документации проекта следует определять, основываясь на практическом опыте экспертов, с учетом результатов прогнозных оценок.

**Разработка документации** (четвертая работа на рис. 42) выполняется в соответствии с утвержденным планом документирования.

При **проверке документации** (пятая работа на рис. 42) должны быть выполнены проверки ее первой и второй редакций, а также гранок документации.

Проверка первой редакции предназначена для контроля технической правильности, полноты документации и ее соответствия плану документирования.

Вторая проверка предназначена для контроля правильности внесения в документацию всех изменений, указанных заказчиком при первой проверке.

В подготовленные гранки документации вносятся изменения, указанные при второй проверке.

Шестой работой на рис. 42 является **тестирование документации на практичность**. Оно может быть использовано для оценки практичности документации в соответствии с рекомендациями стандарта *ГОСТ Р ИСО/МЭК 12119–2000* [4].

Требуемый уровень тестирования документации на практичность должен быть определен в плане документирования. Минимально должно быть проведено одно тестирование документации, используемой для выпускаемой версии ПС. Если тестирование документации запланировано до завершения разработки ПС, то ее следует проводить на наиболее близких к окончательной версии прототипах документации и ПС.

При проведении тестирования документации необходимо проверить соответствие документов конкретному ПС.

Условия тестирования документации на практичность должны быть описаны в плане документирования. Они должны включать:

- этапы ЖЦ разработки, на которых должно проводиться тестирование;
- цели тестирования;
- используемые показатели (например, время реакции задач – время, затраченное пользователями на выполнение действий, описанных в документе);
- среду тестирования;
- число и тип привлекаемых пользователей;
- процедуру описания результатов тестирования и рекомендаций по ним;
- процедуру обеспечения реализации рекомендаций по результатам тестирования;
- процедуру доведения результатов тестирования до всего персонала разработчиков документации и заказчика;
- обязанности персонала разработчиков документации, участвующего в тестировании;
- процедуру определения необходимости последующего тестирования.

**Тиражирование и распространение документации** (седьмая работа на рис. 42) производится документатором в соответствии с планом документирования.

В плане документирования должны быть определены **контроль изменений и сопровождение документации**. При этом должны быть предусмотрены четыре типа изменений документации:

- *функциональные изменения данной версии* – изменения функции ПС, внесенные при разработке документации и отраженные в опубликованной документации;
- *функциональные изменения последующей версии* – изменения функции ПС, внесенные при разработке документации и не отраженные в опубликован-

ной документации, но подлежащие учету в последующей редакции документов;

- *изменения ПС после публикации* – изменения функций ПС после издания данной документации;
- *изменения документа после публикации* – изменения в опубликованной документации, обусловленные изменениями ПС или обнаружением погрешностей в данной документации.

Необходимо выполнять проектирование документации так, чтобы допустить внесение в нее изменений всех четырех видов. Для этого необходимо предусмотреть процедуру внесения в документ каждого типа изменений.

*Приложение В* стандарта *ИСО/МЭК 15910* содержит краткие рекомендации по адаптации и практическому использованию положений данного стандарта в конкретном договоре.

## **6.4. Содержание плана документирования**

*План документирования* должен охватывать следующие вопросы (но не ограничиваться ими):

- рабочее наименование, назначение, область применения и ограничения по использованию планируемой документации;
- спецификация стиля;
- определение аудитории пользователей;
- целевое назначение планируемой документации;
- содержание (план-проспект) документации с оценкой ее постраничного объема и соответствующие уточнения для других машинных носителей документации;
- номенклатура поставки – число печатных копий, наличие электронных копий, форматы дисков и файлов (включая версии программных средств);
- установление собственника авторских прав на документацию и любых других прав собственности;
- обеспечение перевода документации на другие языки (при необходимости);
- уровни (грифы) секретности и конфиденциальности документации (при необходимости);
- процедуры и проверки, могущие влиять на процесс разработки документации, включая (при необходимости) хранение, поиск, резервирование, передачу и оценку качества документации;
- методы и средства производства (тиражирования) документации;
- структура коллектива разработчиков документации и, возможно, план выбора данной структуры;
- взаимосвязи проекта документации;
- почасовая загрузка и зарплата разработчиков документации;
- требования к проектным ресурсам (включая информационные и прочие

ресурсы, предоставляемые заказчиком) и срокам их представления;

- метод передачи документатору информации об изменениях программного средства в процессе его разработки;
- планы контроля изменений и сопровождения документации (при необходимости);
- планы проверки документации после ее создания;
- календарное планирование (графики) документирования по контрольным точкам, включая (при необходимости):
  - утверждение плана документирования;
  - подготовку, проверку и корректировку проекта каждого документа;
  - тестирование на практичность;
  - подготовку оригинала фотошаблонов;
  - распечатку, переплетение и распространение документации.

При необходимости каждый из пунктов плана должен быть описан для каждого элемента документации.

В *Приложении С* стандарта *ИСО/МЭК 15910* представлен образец плана документирования, содержащий следующие разделы: область применения и ограничения, оформление и стиль описания, аудитория, проект содержания документации, номенклатура поставки, авторские права, транспортирование, процесс разработки и контроль, тиражирование, коллектив разработчиков документации, ресурсы, тестирование на практичность, график работ.

## **6.5. Требования к содержанию спецификации стиля документации**

В стандарте *ИСО/МЭК 15910* отражено содержание спецификации стиля документации.

Спецификация стиля должна содержать правила оформления документации различных типов. Например, для бумажных документов это может быть компоновка документа, заголовки, шрифты, перечень вводных и вспомогательных материалов, схемы нумерации страниц, таблиц, рисунков и т.п. Для электронных документов это может быть подсветка, использование цвета, шрифты, заголовки, навигация и т.п. Стандарт *ИСО/МЭК 15910* не определяет состава конкретных требований к оформлению документов, а определяет лишь содержание и диапазоны требований.

*Для бумажных документов* в спецификации стиля определены следующие *вводные и вспомогательные материалы*:

- титульный лист;
- информация о гарантиях, авторских правах и торговой марке;
- содержание;

- список рисунков и таблиц;
- приложения;
- словарь (гlossарий);
- перечень обозначений и сокращений;
- указатель.

**Для электронных документов** в спецификации стиля предусмотрены следующие *типы справочной информации*:

- *контекстная справка (context help)* – информация о текущем поле, в котором находится курсор или высвечена программа, включая ее необходимое или предметное содержание, назначение и потенциальное влияние на работу ПС;
- *расширенная справка (extended help)* – информация о текущем экране или окне, включая ее назначение и требуемый режим использования;
- *справка о клавишах (keys help)* – информация о применении клавиатуры, функциональном назначении клавиш и их обозначении;
- *справка о справках (help to help)* – информация об использовании справочной системы в целом;
- *справка о сообщении (message help)* – информация о возможных действиях пользователя в ответ на конкретные системные сообщения (например, сообщения об ошибках);
- *терминологическая справка (reference phrase help)* – определения конкретных элементов, связей с конкретной тематикой и расшифровка обозначений и сокращений;
- *понятийная справка (intelligent help)* – справочная информация, выдаваемая системой для предупреждения пользователя об ошибках.

**Для электронных документов** в спецификации стиля предусмотрены также следующие *типы диалоговой (оперативной) документации*:

- *руководства или рекомендации пользователя (user guide or reference)* – описывают процедуры по применению продукта;
- *командные справки (command reference)* – определяют синтаксис, результаты и целевое назначение каждой команды (только для команд управления системой);
- *рекомендации по сообщению (message reference)* – определяют действия пользователя в ответ на конкретные системные сообщения (например сообщения об ошибках);
- *административная информация (administration information)* – информация для администратора системы о конфигурации и защите системы, при необходимости – о ее инсталляции.

Дополнительные рекомендации по содержанию спецификации стиля документов приведены в *Приложении Н* стандарта ИСО/МЭК 15910.

## 7. СТАНДАРТ ISO/IEC 15288

### 7.1. Общие сведения

В последние годы наблюдалось существенное развитие рынка услуг системного проектирования и связанных с ним продуктов. В то же время данное развитие осложнялось отсутствием стандартов в области ЖЦ систем, базирующихся на методах системного проектирования и учитывающих аппаратные и программные компоненты систем в интегрированной форме.

Действующий международный стандарт *ИСО/МЭК 12207:1995* содержит только те аспекты описания системы, которые необходимы для понимания сути их программных компонентов или услуг. Данный стандарт разработан с некоторой степенью расширения для применения в процессе системной инженерии. Он определяет жизненный цикл системы в целом, однако охватывает такие процессы, как разработка, эксплуатация и сопровождение системы только в части ее программных средств.

С учетом этого был разработан и в 2002 г. введен в действие международный стандарт *ISO/IEC 15288:2002 – Системная инженерия – Процессы жизненного цикла системы* [29]. Целью его разработки является определение общей структуры описания жизненного цикла систем.

В 2003 г. опубликован международный стандарт *ISO/IEC TR 19760:2003 – Системная инженерия – Руководство по использованию ISO/IEC 15288 (Процессы жизненного цикла системы)* [44], содержащий рекомендации по практическому применению стандарта *ISO/IEC 15288:2002* в условиях реализации конкретных проектов создания систем.

Стандарт *ISO/IEC 15288* устанавливает всестороннюю и интегрированную структуру для управления полным жизненным циклом систем, представляющих собой совокупность аппаратных средств, программных средств, персонала (ручных операций) и процессов. Стандарт определяет набор процессов ЖЦ и связанную с ними терминологию. Эти процессы могут применяться на любом уровне иерархии разработки систем.

Стандарт *ISO/IEC 15288* может быть использован в большинстве отраслей современной промышленности, связанных с разработкой и использованием систем (например, в аэрокосмической промышленности, телекоммуникациях, военной промышленности, информационных технологиях и т.п.).

Применение данного стандарта обеспечивает основу для целостного подхода к программному обеспечению и системному проектированию; профессиональной разработки систем; повышения качества разрабатываемых в системном проекте продуктов; существенного сокращения рисков системных проектов; использования моделей жизненных циклов, разделённых на стадии.



## 7.2. Процессы жизненного цикла систем

В соответствии со стандартом *ISO/IEC 15288* жизненный цикл систем имеет трехуровневую иерархическую структуру (рис. 43).

Основу жизненного цикла составляет набор **процессов**, каждому из которых соответствует общая **цель** его выполнения. Итогом процесса являются **результаты**, связанные с успешным достижением его цели. Каждый процесс с целью его структуризации разделен на набор **работ**, необходимых для получения результатов процесса. *Общее число* процессов в ЖЦ системы равно 25, результатов – 123, работ – 208.



Рис. 43. Иерархическая структура жизненного цикла систем в соответствии со стандартом *ISO/IEC 15288*

Представление процессов жизненного цикла посредством их результатов позволяет согласовать положения стандарта *ISO/IEC 15288* с положениями серии стандартов *ISO/IEC 15504-1-9* [30 – 38], широко используемых в настоящее время для оценки качества процессов ЖЦ. Структура жизненного цикла стандарта *ISO/IEC 15288* может быть использована в качестве эталонной модели ЖЦ систем при оценке процессов жизненного цикла конкретных проектов, выполняемой в соответствии с рекомендациями стандартов серии *ISO/IEC 15504*.

В соответствии со стандартом *ISO/IEC 15288* процессы жизненного цикла систем делятся на **четыре группы**:

- процессы согласования (The Agreement Processes);
- процессы предприятия (The Enterprise Processes);
- процессы проекта (The Project Processes);
- технические процессы (The Technical Processes).

На рис. 44 изображены процессы жизненного цикла системы с учетом их распределения по группам.

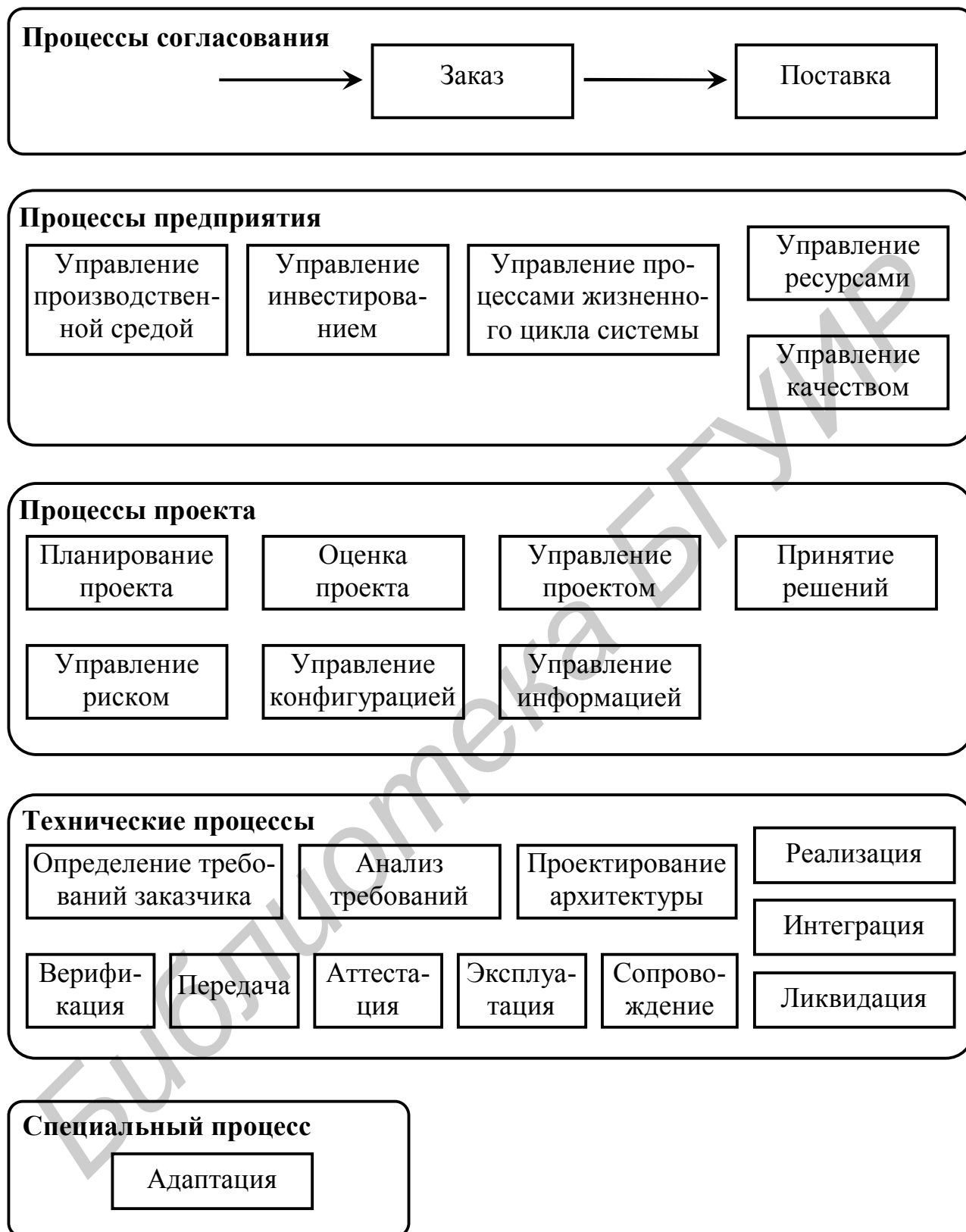


Рис. 44. Процессы жизненного цикла системы по стандарту ISO/IEC 15288:2002

**Процессы согласования** – это процессы, с помощью которых организуются взаимосвязи между инициатором проекта, поставщиком результатов проекта и непосредственным исполнителем проекта. Данные процессы предназначены для инициирования проекта, согласования условий и результатов его выполнения. К группе процессов согласования относятся *два процесса*:

- заказ (The Acquisition Process);
- поставка (The Supply Process).

**Процессы предприятия** – это процессы, которые управляют деятельностью предприятия в целом, не связанной с выполнением конкретного проекта. Данные процессы предназначены для управления теми сторонами работы предприятия, которые влияют на жизненный цикл системных проектов.

В состав группы процессов предприятия входят *пять процессов*:

- управление производственной средой (The Enterprise Environment Management Process);
- управление инвестированием (The Investment Management Process);
- управление процессами жизненного цикла системы (The System Life Cycle Processes Management Process);
- управление ресурсами (The Resource Management Process);
- управление качеством (The Quality Management Process).

**Процессы проекта** – это процессы, управляющие деятельностью предприятия, связанной с выполнением конкретного проекта.

В группу процессов проекта входят *семь процессов*:

- планирование проекта (The Project Planning Process);
- оценка проекта (The Project Assessment Process);
- управление проектом (The Project Control Process);
- принятие решений (The Decision-Making Process);
- управление риском (The Risk Management Process);
- управление конфигурацией (The Configuration Management Process);
- управление информацией (The Information Management Process).

**Технические процессы** – это процессы, непосредственно связанные с разработкой, эксплуатацией и сопровождением системы.

В состав группы технических процессов входят *одиннадцать процессов*:

- определение требований заказчика (The Stakeholder Requirements Definition Process);
- анализ требований (The Requirements Analysis Process);
- проектирование архитектуры (The Architectural Design Process);
- реализация (The Implementation Process);
- интеграция (The Integration Process);
- верификация (The Verification Process);
- передача (The Transition Process);
- аттестация (The Validation Process);
- эксплуатация (The Operation Process);

- сопровождение (The Maintenance Process);
- ликвидация (The Disposal Process).

В **специальную группу процессов** входит процесс адаптации (The Tailoring Process), предназначенный для практического приспособления остальных процессов ЖЦ к условиям реализации конкретного проекта.

Процессы, входящие в группу технических процессов ЖЦ систем (см. рис. 44), по названиям в основном соответствуют работам процесса разработки ЖЦ программных средств, определенным в стандарте *ИСО/МЭК 12207* (см. рис. 7 пособия). Различие заключается в том, что данные процессы ориентированы как на программные средства, так и на технические средства и ручные операции систем.

В *процессе определения требований заказчика* (см. рис. 44) разрабатывается концепция системы, формулируются общие требования к ней.

В *процессе анализа требований* анализируется предметная область, выделяются деловые процессы и потоки информации. На основании этого разрабатываются требования к системе.

В *процессе проектирования архитектуры* выполняется разделение системы на технические объекты, программные объекты и ручные операции, требования к системе распределяются между ними.

В *процессе реализации* осуществляется параллельная разработка технических и программных средств и ручных операций системы. Укрупненная структура данного процесса представлена на рис. 45.

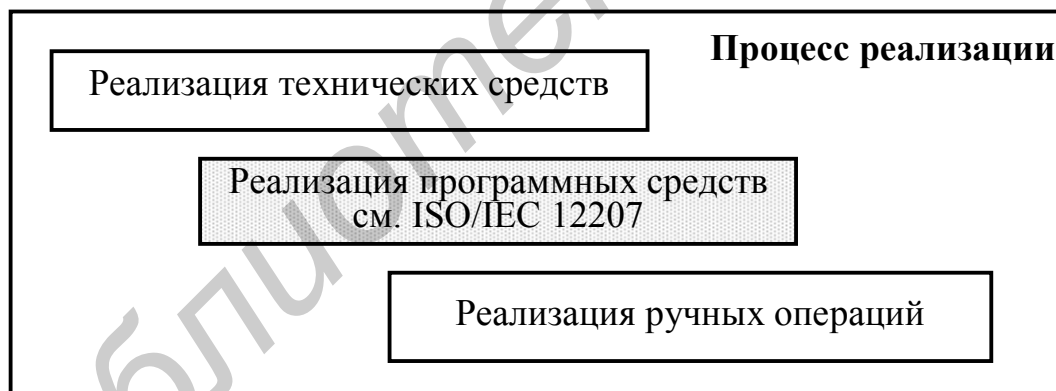


Рис. 45. Укрупненная структура процесса реализации стандарта ISO/IEC 15288

На данном рисунке проиллюстрирована связь стандартов *ISO/IEC 15288* и *ISO/IEC 12207*: процесс реализации из стандарта *ISO/IEC 15288* в части программных средств системы выполняется в соответствии с процессом разработки стандарта *ISO/IEC 12207* (работы 5.3.4 – 5.3.9, см. рис. 7 пособия).

В *процессе интеграции* осуществляется сборка технических и программных средств системы и ручных операций.

В *процессе верификации* выполняется проверка результатов сборки систе-

мы на соответствие требованиям.

В процессе передачи система устанавливается в реальную среду эксплуатации.

В процессе аттестации выполняется проверка системы в условиях эксплуатации на соответствие требованиям.

В процессе эксплуатации осуществляется использование системы в реальных условиях.

В процессе сопровождения выполняются необходимые модификации системы.

В процессе ликвидации осуществляется прекращение использования и уничтожение системы.

Определение в стандарте *ISO/IEC 15288* процессов ЖЦ с помощью их целей представляет собой удобную основу для использования моделей жизненного цикла систем, базирующихся на некоторых периодах ЖЦ (например стадиях, этапах или фазах), представляющих собой объединение нескольких процессов или работ ЖЦ по какому-либо признаку (см. подразд. 4.5 пособия).

В Приложении В стандарта *ISO/IEC 15288* приведен пример возможного объединения процессов и работ ЖЦ в стадии. Перечень данных стадий и их содержание отражает табл. 35.

Таблица 35

Возможные стадии жизненного цикла систем

Стадия	Описание
Концепция (Concept Stage)	Анализ потребностей, определение концепций и выработка решений
Разработка (Development Stage)	Разработка продуктов проекта
Производство (Production Stage)	Изготовление продуктов и их сборка в систему
Использование (Utilization Stage)	Эксплуатация системы в заданной среде
Поддержка (Support Stage)	Сопровождение и поддержка системы
Изъятие (Retirement Stage)	Изъятие, ликвидация и архивация системы

Таким образом, применение стандарта *ISO/IEC 15288* помогает определить соответствующую проекту модель жизненного цикла системы, что является ключевым фактором эффективной разработки, производства и использования систем.

# ВОПРОСЫ ДЛЯ САМОПРОВЕРКИ К ЧАСТИ I

## Раздел 1

1. Дайте определение жизненного цикла (ЖЦ) программного средства (ПС) или системы.
2. Дайте определение модели ЖЦ ПС или системы.
3. Что такое аттестация ПС?
4. Что такое верификация ПС?
5. Опишите понятие базовой линии.
6. Дайте определение элемента конфигурации.

## Раздел 2

1. Назовите базовый стандарт в области ЖЦ ПС и систем.
2. Что подразумевается под системой при стандартизации ЖЦ?
3. Определите иерархическую структуру ЖЦ ПС.
4. На какие группы делятся процессы ЖЦ и в соответствии с чем?
5. Назовите основные стороны, участвующие в ЖЦ ПС и систем.
6. Перечислите процессы ЖЦ в каждой группе.
7. Опишите сущность каждого из основных процессов ЖЦ.
8. Назовите процессы ЖЦ, влияющие на управление качеством ПС.
9. Перечислите работы процесса разработки и опишите их содержание.
10. В каком процессе ЖЦ разрабатывается план управления проектом?
11. Перечислите содержание плана управления проектом.
12. В каком процессе ЖЦ разрабатываются планы выполнения процессов?
13. Опишите содержание планов выполнения процессов.
14. Перечислите состав требований к системе и назовите критерии их оценки.
15. Перечислите состав требований к ПС и назовите критерии их оценки.
16. Определите структуру работ по проектированию ПС и их результаты.
17. Назовите критерии оценки результатов каждой работы процесса разработки.
18. Назовите основные документы процесса разработки.
19. Опишите сущность каждого из вспомогательных процессов ЖЦ.
20. Опишите сущность каждого из организационных процессов ЖЦ.
21. Что такое инфраструктура процесса?
22. Опишите сущность процесса адаптации положений стандарта *ИСО/МЭК 12207* к условиям конкретного проекта.
23. Какие характеристики проекта влияют на адаптацию?

## Раздел 3

1. Какие общие изменения внесены в стандарт *ISO/IEC 12207:1995* Дополнением *ISO/IEC 12207:1995/Amd.1:2002*?
2. Назовите новые процессы ЖЦ ПС и опишите их назначение.
3. Назовите процессы ЖЦ с измененной структурой.
4. Опишите различия в структурах нового и старого процессов разработки.
5. Опишите структуру нового процесса управления.

6. Опишите сущность и структуру расширенного описания процесса заказа.

#### **Раздел 4**

1. Перечислите системные и программные работы процесса разработки.
2. Перечислите основные типы документов ЖЦ ПС.
3. Назовите способы реализации управления качеством в *ИСО/МЭК 12207*.
4. Опишите основные шаги стратегии практического применения стандарта *ИСО/МЭК 12207*.
5. Перечислите факторы, влияющие на практическое применение *ИСО/МЭК 12207* в конкретных проектах.
6. Нарисуйте и опишите типовую модель ЖЦ системы.
7. Опишите каждую из фундаментальных моделей ЖЦ ПС и системы.
8. Опишите модели ЖЦ, использующие прототипирование.

#### **Раздел 5**

1. На каких документах основана разработка стандарта *ИСО/МЭК ТО 16326*?
2. Что такое проект?
3. Чем отличаются программные проекты от других проектов?
4. Назовите виды управления проектом, определенные в Руководстве РМВОК.
5. Что такое область проекта?
6. Опишите сущность каждой из работ процесса управления.
7. Опишите содержание планов выполнения процессов.

#### **Раздел 6**

1. Назовите виды документации пользователя.
2. Что такое аудитория?
3. Опишите структуру процесса документирования, определенного в стандарте *ИСО/МЭК 15910*.
4. Опишите последовательность проектирования документации, установленную в *ИСО/МЭК 15910*.
5. Назовите и опишите методы оценки проекта документации.
6. Что такое тестирование документации на практичность?
7. Назовите типы изменений документации.
8. Перечислите основные разделы плана документирования.
9. Перечислите типы электронной справочной информации.

#### **Раздел 7**

1. Определите иерархическую структуру ЖЦ системы по *ISO/IEC 15288:2002*.
2. Перечислите группы, на которые делятся процессы ЖЦ систем.
3. Дайте определение каждой из групп процессов ЖЦ системы.
4. Назовите процессы, входящие в состав каждой из групп ЖЦ системы.
5. Поясните связь между стандартами *ISO/IEC 12207:1995* и *ISO/IEC 15288:2002*.
6. Приведите пример объединения процессов ЖЦ систем в стадии.

# **ЧАСТЬ II**

## **СТАНДАРТИЗАЦИЯ ОЦЕНКИ КАЧЕСТВА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

Библиотека ЮНЕСКО



## 8. КАЧЕСТВО ПРОГРАММНЫХ СРЕДСТВ. ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ

В настоящее время компьютеры находят все более широкое применение во всех сферах жизнедеятельности человека. От правильности их работы во многом зависят результаты труда и безопасность тысяч людей. В этой связи в последние годы во всем мире особое внимание уделяется стандартизации прогнозирования, оценки и управления качеством ПС на протяжении всего их ЖЦ.

На процесс разработки и деятельность по оценке качества ПС оказывают влияние следующие *обобщенные показатели ПС* [58]:

- область применения и назначение ПС;
- тип решаемых задач;
- объем и сложность ПС;
- необходимый состав и требуемые значения характеристик качества ПС и величина допустимого ущерба из-за их недостаточного качества;
- степень связи решаемых задач с реальным масштабом времени или допустимой длительностью ожидания результатов решения задачи;
- прогнозируемые значения длительности эксплуатации и перспектива создания множества версий ПС;
- предполагаемый тираж производства и применения ПС;
- степень необходимой документированности ПС.

Существует ряд международных, межгосударственных и национальных стандартов, посвященных вопросам управления качеством ПС, включая планирование, обеспечение и контроль качества ПС. Основным компонентом контроля качества ПС и основой для его обеспечения является оценка качества.

В данной части пособия используются следующие **основные термины**, соответствующие определениям *ГОСТ 28806–90, СТБ ИСО/МЭК 9126–2003, ISO/IEC 9126–1–4:2001–2004, ISO/IEC 14598–1:1999* [3, 14, 49 – 52, 21].

*Атрибут (attribute)*: измеримое физическое или абстрактное свойство продукта. Атрибуты могут быть внешними или внутренними.

*Внешнее качество (external quality)*: степень, в которой продукт удовлетворяет установленные и подразумеваемые потребности при использовании в заданных условиях.

*Внешняя мера (external measure)*: косвенная мера продукта, полученная из мер поведения системы, частью которой он является. Внешние меры могут использоваться для оценки атрибутов качества промежуточных продуктов ближе к конечным целям проекта.

*Внутреннее качество (internal quality)*: полный набор атрибутов продукта,

определяющих его способность удовлетворять установленные и подразумеваемые потребности при использовании в заданных условиях.

*Внутренняя мера (internal measure)*: собственная мера продукта, прямая или косвенная.

*Измерение (measurement)*: использование метрики для присвоения атрибуту продукта значения (числа или категории) из шкалы.

*Индикатор (indicator)*: мера, которая может использоваться для оценки или прогнозирования другой меры.

*Качество (quality)*: совокупность характеристик ПП, относящаяся к его способности удовлетворять установленные и подразумеваемые потребности.

*Качество в использовании (quality in use)*: степень, в которой программный продукт, используемый заданными пользователями, удовлетворяет их потребности в достижении заданных целей с результативностью, продуктивностью, безопасностью и удовлетворенностью в заданном контексте использования.

*Контекст использования (context of use)*: пользователи, задания, среда (аппаратное обеспечение, программное обеспечение и материалы), а также физические и социальные среды, в которых используется данное ПС.

*Критерий оценки качества (software quality assessment criteria)*: совокупность принятых в установленном порядке правил и условий, с помощью которых устанавливается приемлемость общего качества программного продукта.

*Мера (measure)*: число или категория, присваиваемая атрибуту продукта путем измерения.

*Мера косвенная (indirect measure)*: мера атрибута, которая получена из мер одного или большего числа других атрибутов.

*Мера прямая (direct measure)*: мера атрибута, которая не зависит от меры любого другого атрибута.

*Метрика (metric)*: определенный метод и шкала измерения. Метрики могут быть внутренними, внешними или метриками качества в использовании; прямыми или косвенными. Метрики включают методы для категоризации качественных данных (данных, которые нельзя измерить количественно).

*Модель качества (quality model)*: набор характеристик и связей между ними, обеспечивающий основу для определения требований к качеству и для оценки качества.

*Отказ (failure)*: прекращение способности продукта выполнять требуемую функцию или его неспособность работать в пределах заданных ограничений.

*Оценка качества (quality evaluation)*: Систематическое исследование степени, в которой продукт способен к выполнению указанных требований.

*Оценочный модуль (evaluation module)*: пакет технологии оценивания для конкретной характеристики или подхарактеристики качества ПС.

*Ошибка (fault)*: некорректный шаг, процесс или определение данных в программе.

*Подразумеваемые потребности (implied needs)*: потребности, которые не были установлены, но являются действительными потребностями при использовании продукта в конкретных условиях.

*Подхарактеристика качества ПС (software quality subcharacteristic)*: это характеристика качества программного средства, входящая в состав другой характеристики качества.

*Показатель качества ПС (software quality feature)* – признак, определяющий свойство программного средства, которое может быть соотнесено с некоторой характеристикой качества.

*Промежуточный программный продукт (intermediate software product)*: продукт процесса разработки программного обеспечения, который используется в качестве входных данных для другой стадии процесса разработки программного обеспечения. Промежуточный продукт может также быть конечным продуктом.

*Ранжирование (rating)*: действие по отнесению измеренного значения к соответствующему уровню ранжирования.

*Уровень качества функционирования (уровень пригодности, level of performance)*: степень удовлетворения потребности, представленная конкретным набором значений характеристик качества.

*Уровень ранжирования (уровень оценки, rating level)*: точка на порядковой шкале, которая используется для категоризации шкалы измерения. Уровень оценки позволяет ранжировать программное обеспечение в соответствии с установленными или подразумеваемыми потребностями. Соответствующие уровни ранжирования могут быть связаны с различными точками зрения на качество, например, пользователей, администраторов или разработчиков.

*Характеристика качества ПС (software quality characteristic)*: набор свойств программного средства, с помощью которых описывается и оценивается его качество.

*Шкала (scale)*: набор значений с определенными свойствами.

При оценке качества используются следующие *типы шкал*:

- *номинальная* – соответствует набору категорий; классифицирует программы по признаку наличия или отсутствия некоторого свойства без учета градаций (например «да», «нет»);
- *порядковая (упорядоченная)* – соответствует упорядоченному набору делений шкалы; позволяет ранжировать свойства путем сравнения с опорными значениями; имеет небольшое количество делений (например, шкала с четырьмя градациями «отлично», «хорошо», «удовлетворительно», «неудовлетворительно», с двумя градациями «удовлетворительно», «неудовлетворительно»);
- *интервальная* – соответствует упорядоченной шкале с равноудаленными делениями; обычно содержит достаточно большое количество делений с количественными значениями (например шкала с делениями 0, 1, 2, ..., 10);
- *относительная* – соответствует упорядоченной шкале с равноудаленными делениями, оцененными в относительных единицах относительно некоторой абсолютной величины (обычно в диапазоне от 0 до 1).

Два первых типа шкал применяются для оценки качественных атрибутов ПС, которые нельзя измерить количественно, и для ранжирования измеренных значений, третий и четвертый типы – для оценки количественных атрибутов.

## 9. СТАНДАРТИЗАЦИЯ КАЧЕСТВА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ В РЕСПУБЛИКЕ БЕЛАРУСЬ

### 9.1. Общие сведения о стандартах в области оценки качества, действующих на территории Республики Беларусь

В настоящее время в области оценки качества ПС на территории Республики Беларусь действуют следующие основные стандарты:

- стандарт СССР *ГОСТ 28806–90. Качество программных средств. Термины и определения* [3];
- межгосударственный стандарт стран СНГ *ГОСТ 28195–99. Оценка качества программных средств. Общие положения* [2];
- национальный стандарт Беларуси *СТБ ИСО/МЭК 9126–2003. Информационные технологии. Оценка программной продукции. Характеристики качества и руководства по их применению* [14].

Стандарт *СТБ ИСО/МЭК 9126–2003* представляет собой аутентичный перевод международного стандарта *ISO/IEC 9126:1991* [48]. Данный стандарт под обозначением *ГОСТ Р ИСО/МЭК 9126–93* действует на территории России с 1994 г. [11].

В стандарте *ГОСТ 28806–90* даются основные термины и определения, принятые в области обеспечения качества программного обеспечения.

*ГОСТ 28195–99* определяет *оценку качества* программного средства как совокупность операций, включающих выбор номенклатуры характеристик качества оцениваемого программного средства, определение значений этих характеристик и сравнение их с базовыми значениями. В соответствии с данным стандартом оценка качества должна проводиться применительно ко всем работам ЖЦ ПС при планировании характеристик качества ПС, контроле качества в процессе разработки, проверке эффективности модификации ПС в процессе сопровождения.

*Основными задачами*, решаемыми при оценке качества программного средства, по *ГОСТ 28195–99* являются:

- 1) планирование номенклатуры характеристик и показателей качества;
- 2) планирование уровня качества;
- 3) выбор методов контроля показателей качества;

- 4) контроль значений показателей качества в процессе ЖЦ ПС;
- 5) выбор базовых образцов по подклассам и группам;
- 6) принятие решения о соответствии реальных значений показателей качества установленным требованиям.

Под **показателем качества продукции** в *ГОСТ 28195–99* подразумевается количественная характеристика одного или нескольких свойств продукции, составляющих ее качество, рассматриваемая применительно к определенным условиям ее создания и эксплуатации или потребления.

Обеспечение и оценка качества ПС выполняются в течение жизненного цикла программных средств и неразрывно связаны с положениями стандарта *ИСО/МЭК 12207*, рассмотренными в разд. 2 пособия.

## **9.2. Классификация методов определения показателей качества программного средства**

Стандарт *ГОСТ 28195–99* и его предыдущая версия *ГОСТ 28195–89* [2, 1] классифицируют методы определения показателей качества ПС следующим образом:

- *по способам получения информации о показателе качества:*
  - измерительный;
  - регистрационный;
  - органолептический;
  - расчетный;
- *по источникам получения информации о показателе качества:*
  - экспертный;
  - социологический;
  - традиционный.

**Измерительный метод** – это метод получения информации о свойствах и характеристиках ПС путем измерений с помощью инструментальных средств (например, так может определяться количество операторов в программе, количество выполненных операторов, количество операндов, время выполнения программы при определенных наборах исходных данных и т.д.).

**Регистрационный метод** – это метод получения информации о свойствах и характеристиках ПС во время его испытания или функционирования, когда регистрируются некоторые события (например количество сбоев и отказов).

**Органолептический метод** – это метод получения информации о свойствах и характеристиках ПС, основанный на восприятии органов чувств (зрения и слуха) человека. Так могут определяться, например, свойства ПС, связанные с удобством его использования.

**Расчетный метод** – это метод получения информации о свойствах и ха-

рактических характеристиках ПС, основанный на использовании эмпирических и теоретических зависимостей (на ранних этапах разработки), статистических данных, накапливаемых при испытаниях, эксплуатации и сопровождении ПС. Так может определяться, например, точность вычислений.

**Экспертный метод** – это метод получения информации о свойствах и характеристиках ПС на основании мнений группы экспертов–специалистов, компетентных в решении данной задачи. Экспертный метод применяется в том случае, когда невозможно или слишком трудоемко выполнить оценку показателей качества с помощью других методов. Данным методом рекомендуется определять, например, показатели понимаемости и осваиваемости ПС.

**Социологический метод** – это метод получения информации о свойствах и характеристиках ПС на основе обработки специальных анкет-опросников. Так могут определяться, например, отдельные показатели удобства использования.

**Традиционный метод** – это метод получения информации о свойствах и характеристиках ПС на основе непосредственного наблюдения за их функционированием в процессе работы. Так могут определяться, например, некоторые из показателей функциональности и удобства использования.

### **9.3. Иерархическая модель оценки качества программного средства**

Стандарты *ГОСТ 28806–90*, *ГОСТ 28195–99*, *СТБ ИСО/МЭК 9126–2003* [3, 2, 14] регламентируют выполнение оценки качества ПС и систем на основе **иерархической модели качества**. В соответствии с данной моделью совокупность свойств, отражающих качество программного средства, представляется в виде многоуровневой структуры. Характеристики на первом (верхнем) уровне соответствуют основным свойствам ПС. Характеристики каждого уровня оцениваются посредством характеристик последующих уровней.

Стандарты *ГОСТ 28806–90*, *СТБ ИСО/МЭК 9126–2003* определяют первые два уровня иерархической модели качества. При этом номенклатура характеристик первого уровня является **обязательной**, а номенклатура характеристик второго уровня (подхарактеристик) – **рекомендуемой**.

Стандарт *ГОСТ 28195–99* определяет **четырёхуровневую** иерархическую модель оценки качества ПС. Номенклатура характеристик и подхарактеристик первых двух уровней является **обязательной**, а номенклатура подхарактеристик третьего и четвертого уровней – **рекомендуемой**.

Вышеназванные стандарты определяют **шесть основных характеристик качества** ПС, находящихся на верхнем уровне модели качества. Следует отметить, что характеристики верхнего уровня, регламентированные *ГОСТ 28806–90* и *СТБ ИСО/МЭК 9126–2003*, соответствуют принятым в настоящее время в мировой практике. В то же время характеристики и подхарактеристики, определенные в *ГОСТ 28195–99*, частично не соответствуют иерархической модели

качества, принятой в международных стандартах.

В стандартах *ГОСТ 28806–90* и *СТБ ИСО/МЭК 9126–2003* определены следующие **основные характеристики качества** ПС (характеристики качества верхнего уровня):

1. **Функциональность (Functionality)** – совокупность свойств ПС, определяемая наличием и конкретными особенностями набора функций, способных удовлетворять заданные или подразумеваемые потребности.

2. **Надежность (Reliability)** – совокупность свойств, характеризующая способность ПС сохранять заданный уровень пригодности в заданных условиях в течение заданного интервала времени.

3. **Удобство использования (практичность, Usability)** – совокупность свойств программного средства, характеризующая усилия, необходимые для его использования, и индивидуальную оценку результатов его использования заданным или подразумеваемым кругом пользователей.

4. **Эффективность (Efficiency)** – совокупность свойств программного средства, характеризующая те аспекты его уровня пригодности, которые связаны с характером и временем использования ресурсов, необходимых при заданных условиях функционирования.

5. **Сопровождаемость (Maintainability)** – совокупность свойств программного средства, характеризующая усилия, которые необходимы для его модификации.

6. **Мобильность (Portability)** – совокупность свойств программного средства, характеризующая приспособленность для переноса из одной среды функционирования в другие.

## **9.4. Метод оценки качества программных средств по ГОСТ 28195–99**

Стандартом *ГОСТ 28195–99* и его предыдущей версией *ГОСТ 28195–89* [2, 1] рекомендован метод интегральной оценки качества программных средств, основанный на иерархической модели качества.

В соответствии с данным методом выбор номенклатуры показателей качества для конкретного программного средства осуществляется с учетом его назначения и требований области применения в зависимости от принадлежности ПС к тому или иному подклассу, определяемому *общесоюзным классификатором продукции (ОКП)*.

В ОКП предусмотрены следующие *подклассы программных средств*:

- 5011 – операционные системы и средства их расширения;
- 5012 – программные средства управления базами данных;
- 5013 – инструментально-технологические средства программирования;
- 5014 – ПС интерфейса и управления коммуникациями;
- 5015 – программные средства организации вычислительного процесса

(например планирования, контроля);

- 5016 – сервисные программы;
- 5017 – ПС обслуживания вычислительной техники;
- 503 – прикладные программы для научных исследований;
- 504 – прикладные программы для проектирования;
- 505 – прикладные программы для управления техническими устройствами и технологическими процессами;
- 506 – прикладные программы для решения экономических задач;
- 509 – прочие программные средства.

Оценка качества ПС производится на всех фазах жизненного цикла.

*ГОСТ 28195–99* базируется на следующих *процессах и фазах жизненного цикла ПС*:

1. Процесс разработки:

- фаза анализа;
- фаза проектирования;
- фаза реализации;
- фаза тестирования;
- фаза изготовления.

2. Процесс применения:

- фаза внедрения;
- фаза эксплуатации;
- фаза сопровождения.

Вышеприведенные фазы представляют собой временные периоды, соответствующие работам, совокупностям работ или процессам ЖЦ ПС, определенным стандартом *ИСО/МЭК 12207* и описанным в разд. 2 пособия.

Оценка качества ПС заключается в выборе номенклатуры показателей, их оценке и сопоставлении с базовыми значениями.

Основу описываемого метода оценки качества составляет *четырёхуровневая иерархическая модель качества*. *ГОСТ 28195–99* предлагает следующую **терминологию** для показателей качества каждого уровня:

*уровень 1* - факторы качества (в терминологии, принятой в международных стандартах, соответствуют характеристикам качества [48, 49]);

*уровень 2* - критерии качества (в международной терминологии – подхарактеристики качества);

*уровень 3* - метрики (соответствует международной терминологии);

*уровень 4* - оценочные элементы или единичные показатели (данный уровень в международных стандартах отсутствует).

Факторы и критерии качества, определенные в стандарте *ГОСТ 28195–99*, приведены на рис. 46.

Для каждого из выбранных факторов качества составляется *четырёхуровневая иерархическая модель*, отражающая взаимосвязь факторов, критериев, метрик и оценочных элементов. Вид данной модели зависит от фазы ЖЦ ПС.



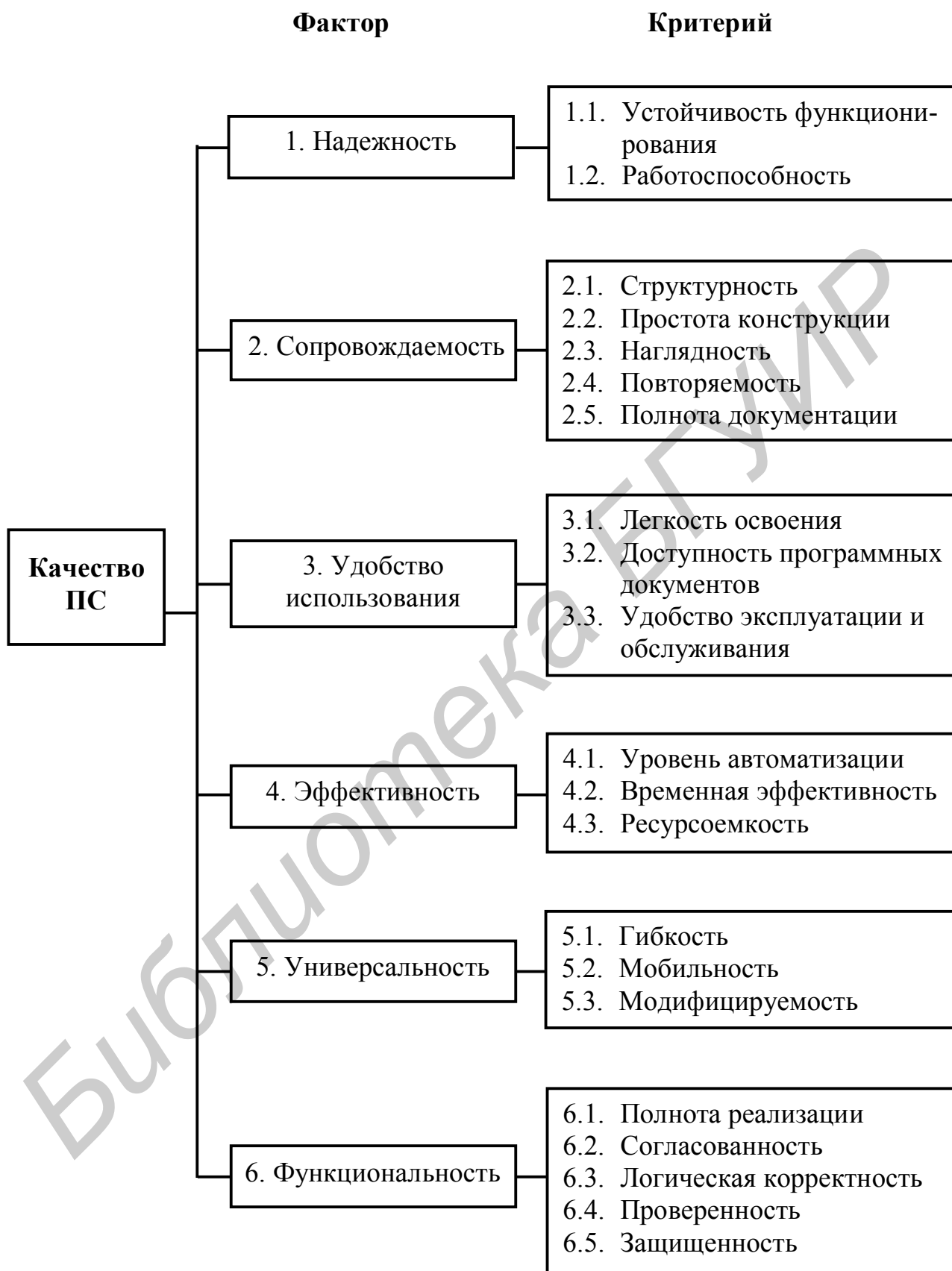


Рис. 46. Факторы и критерии качества программных средств по *ГОСТ 28195–99*

В качестве примера на рис. 47 – 49 приведены три верхних уровня иерархической модели фактора *Надежность* для различных фаз ЖЦ. Номера на данных рисунках соответствуют номерам метрик данного фактора.



Рис. 47. Модель надежности для фазы анализа



Рис. 48. Модель надежности для фазы проектирования

Выбор оценочных элементов в метрике зависит от функционального назначения ПС и формируется с учетом данных, ранее полученных при проведении испытаний ПС и эксплуатации аналогичных программ. Для выбора оценочных элементов *ГОСТ 28195–99* предлагает перечень таблиц, содержащих наименование элемента, метод оценки и применяемость элемента для различных подклассов ПС.

Табл. 36 содержит перечень оценочных элементов для фактора *Надежность*. В данной таблице код оценочного элемента состоит из пяти символов. Первый символ (буква) указывает на принадлежность элемента фактору (в *ГОСТ 28195–99* приняты следующие обозначения факторов: Н – Надежность; С – Сопровождаемость; У – Удобство использования; Э – Эффективность; Г – Универсальность; К – Функциональность). Два следующих символа – номер

метрики, которой принадлежит оценочный элемент (для *Надежности* номера метрик обозначены на рис. 47 – 49). Четвертый и пятый символы – порядковый номер данного оценочного элемента в метрике.

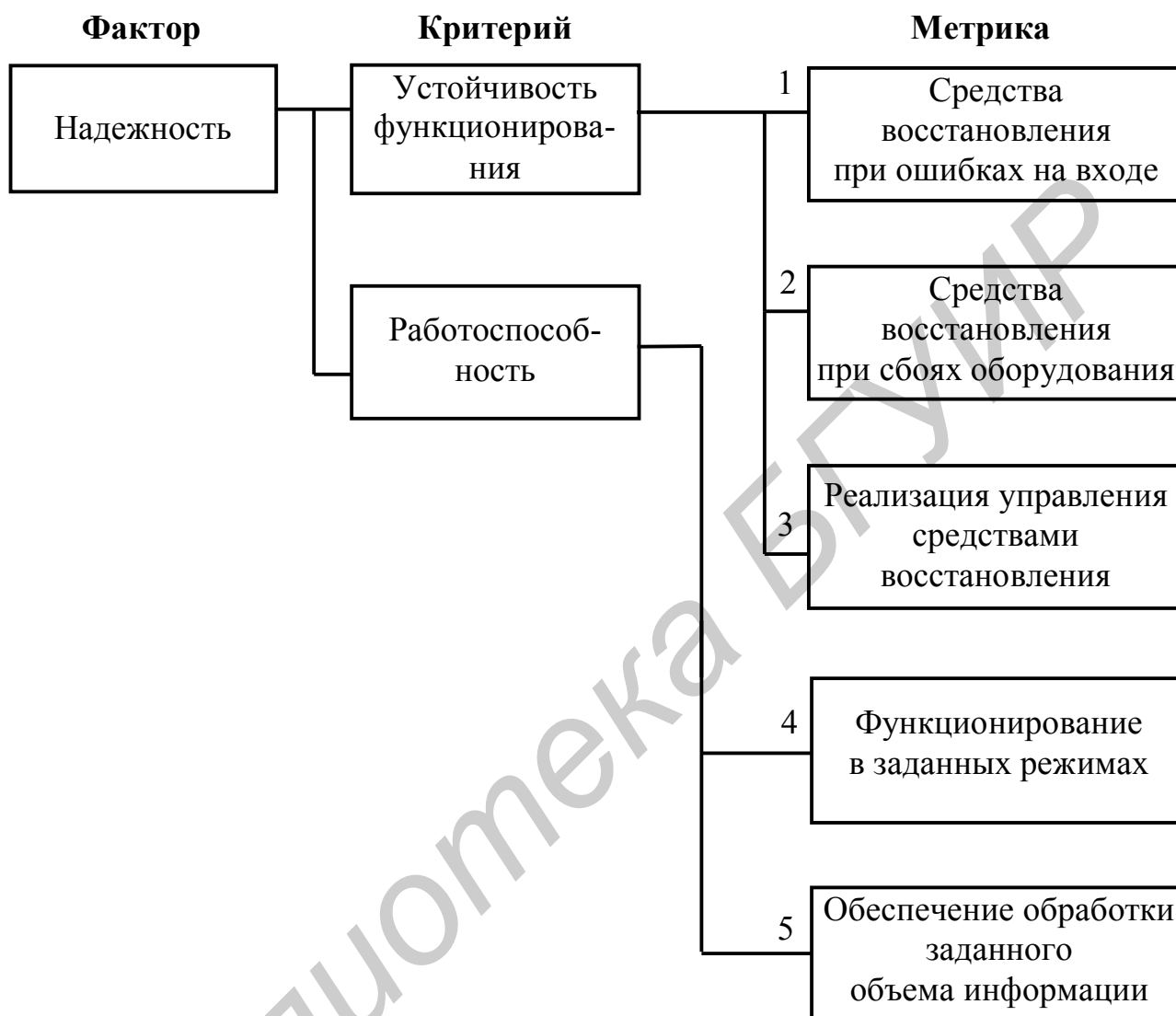


Рис. 49. Модель надежности для фаз реализации, тестирования, изготовления и сопровождения

Таблица 36

Оценочные элементы фактора *Надежность*

Код элемента	Наименование	Метод оценки	Оценка
Н0101	Наличие требований к программе по устойчивости функционирования при наличии ошибок во входных данных	Экспертный	0 – 1

Код элемента	Наименование	Метод оценки	Оценка
H0102	Возможность обработки ошибочных ситуаций	Экспертный	0 – 1
H0103	Полнота обработки ошибочных ситуаций	– " –	0 – 1
H0104	Наличие тестов для проверки допустимых значений входных данных	– " –	0 – 1
H0105	Наличие системы контроля полноты входных данных	– " –	0 – 1
H0106	Наличие средств контроля корректности входных данных	– " –	0 – 1
H0107	Наличие средств контроля непротиворечивости входных данных	– " –	0 – 1
H0108	Наличие проверки параметров и адресов по диапазону их значений	– " –	0 – 1
H0109	Наличие обработки граничных результатов	– " –	0 – 1
H0110	Наличие обработки неопределенностей	– " –	0 – 1
H0201	Наличие требований к программе по восстановлению процесса выполнения в случае сбоя операционной системы, процессора, внешних устройств	– " –	0 – 1
H0202	Наличие требований к программе по восстановлению результатов при отказах процессора, операционной системы	– " –	0 – 1
H0203	Наличие средств восстановления процесса в случае сбоев оборудования	– " –	0 – 1
H0204	Наличие возможности разделения по времени выполнения отдельных функций программ	– " –	0 – 1
H0205	Наличие возможности повторного старта с точки останова	– " –	0 – 1
H0301	Наличие централизованного	– " –	0 – 1

Код элемента	Наименование	Метод оценки	Оценка
	управления процессами, конкурирующими из-за ресурсов		
Н0302	Наличие возможности автоматически обходить ошибочные ситуации в процессе вычисления	Экспертный	0 – 1
Н0303	Наличие средств, обеспечивающих завершение процесса решения в случаях помех	– " –	0 – 1
Н0304	Наличие средств, обеспечивающих выполнение программы в сокращенном объеме в случае ошибок или помех	– " –	0 – 1
Н0305	Показатель устойчивости к искажающим воздействиям $P(Y)$	Регистрационный + Расчетный	$P(Y) = 1 - D/K$ , где: $D$ – число экспериментов, в которых искажающие воздействия приводят к отказу; $K$ – число экспериментов, в которых имитируются искажающие воздействия; $Y$ – вид искажающего воздействия
Н0401	Вероятность безотказной работы $P$	Регистрационный + Расчетный	$P = 1 - Q/N$ , где: $N$ – число экспериментов; $Q$ – число зарегистрированных отказов
Н0501	Оценка по среднему времени восстановления $Q$	Измерительный + Расчетный	$Q = \begin{cases} 1, & \text{если } T_{\text{в}} \leq T_{\text{в доп}}; \\ T_{\text{в доп}}/T_{\text{в}}, & \text{если } T_{\text{в}} > T_{\text{в доп}}, \end{cases}$ <p>где: <math>T_{\text{в}}</math> – среднее время восстановления:</p> $T_{\text{в}} = \frac{1}{N} \sum_{i=1}^N T_{\text{ви}};$ <p><math>N</math> – число восстанов-</p>

Код элемента	Наименование	Метод оценки	Оценка
			лений; $T_{ei}$ – время восстановления после $i$ -го отказа; $T_{e\text{ доп}}$ – допустимое среднее время восстановления
Н0502	Оценка по продолжительности преобразования $i$ -го входного набора данных в выходной $Q_{ni}$	Измерительный + Расчетный	$Q_{ni} = \begin{cases} 1, & \text{если } T_{ni} \leq T_{ni\text{ доп}}; \\ T_{e\text{ доп}} / T_{ei}, & \text{если } T_{ni} > T_{ni\text{ доп}}, \end{cases}$ <p>где <math>T_{ni\text{ доп}}</math> – допустимое время преобразования <math>i</math>-го входного набора данных;  <math>T_{ni}</math> – фактическая продолжительность преобразования <math>i</math>-го входного набора данных</p>

Оценка качества программного средства проводится в следующей последовательности:

1. На фазе анализа проводится выбор показателей и их базовых значений.
2. Для показателей качества на всех уровнях принимается единая шкала оценки (от 0 до 1).
3. В процессе оценки качества на каждом уровне (кроме уровня оценочных элементов) проводится вычисление двух величин:

- абсолютного показателя качества  $P_{ij}$ ;
- относительного показателя качества  $R_{ij}$ ,

где  $j$  – порядковый номер показателя данного уровня для  $i$ -го показателя вышестоящего уровня.

Относительный показатель качества  $R_{ij}$  является функцией показателя  $P_{ij}$  и его базового значения  $P_{ij}^{\bar{o}}$  и определяется по формуле

$$R_{ij} = \frac{P_{ij}}{P_{ij}^{\bar{o}}}.$$

ГОСТ 28195–99 содержит таблицу с базовыми значениями для характеристик качества второго уровня (критериев). Данные значения определяются подклассом программного средства в соответствии с ОКП.

Базовые значения для показателей первого и третьего уровней формируются методом экспертного опроса с учетом назначения ПС или на основании показателей существующих аналогов или расчетного эталонного ПС. Значения базовых показателей ПС должны соответствовать значениям показателей, отражающих современный уровень качества и прогнозируемый мировой уровень.

4. Каждый показатель качества второго и третьего уровней характеризуется двумя параметрами:

- количественным значением;
- весовым коэффициентом  $V_{ij}$ .

Сумма весовых коэффициентов всех показателей некоторого уровня, относящихся к показателю вышестоящего уровня, постоянна и равна 1:

$$\sum_{j=1}^J V_{ij} = 1,$$

где  $J$  – общее количество всех показателей  $j$ -го уровня, относящихся к  $i$ -му показателю вышестоящего уровня, определенных в стандарте.

ГОСТ 28195–99 содержит таблицы, содержащие перечни весовых коэффициентов для характеристик второго и третьего уровней (критериев и метрик). Количественные величины весовых коэффициентов зависят от фазы ЖЦ ПС и подкласса ПС в соответствии с ОКП.

5. Определение усредненной оценки  $m_{kq}$  оценочного элемента по нескольким его значениям (измерениям)  $m_{qt}$  осуществляется по следующей формуле (формула для вычисления значений показателей качества 4-го уровня):

$$m_{kq} = \frac{1}{T} \sum_{t=1}^T m_{qt},$$

где  $k$  – порядковый номер метрики;  $q$  – порядковый номер оценочного элемента;  $T$  – число значений (измерений) оценочного элемента;  $t$  – номер значения оценочного элемента.

6. Итоговая оценка  $k$ -ой метрики  $j$ -го критерия определяется по формуле (формула для вычисления значений показателей качества 3-го уровня):

$$P_{jk}^M = \frac{1}{Q} \sum_{q=1}^Q m_{kq},$$

где  $M$  – признак метрики;  $Q$  – число оценочных элементов, реально используемых при оценке  $k$ -й метрики.

7. Абсолютные показатели  $j$ -го критерия  $i$ -го фактора вычисляются по формуле (формула для вычисления значений показателей качества 2-го уровня):

$$P_{ij}^K = \frac{\sum_{k=1}^n (P_{jk}^M \cdot V_{jk}^M)}{\sum_{k=1}^n V_{jk}^M},$$

где  $n$  – число метрик, относящихся к  $j$ -му критерию, реально используемых при оценке;  $K$  – признак критерия.

8. Относительные значения  $R_{ij}^K$   $j$ -го критерия  $i$ -го фактора  $P_{ij}^K$  по отношению к базовому значению  $P_{ij}^{K_0}$  определяются по формуле

$$R_{ij}^K = \frac{P_{ij}^K}{P_{ij}^{K_0}}.$$

9. Абсолютные и относительные значения  $i$ -го фактора качества определяются по формулам (формулы для вычисления значений показателей качества 1-го уровня)

$$P_i^\Phi = \frac{\sum_{j=1}^N (P_{ij}^K \cdot V_{ij}^K)}{\sum_{j=1}^N V_{ij}^K},$$

$$R_i^\Phi = \frac{\sum_{j=1}^N (R_{ij}^K \cdot V_{ij}^K)}{\sum_{j=1}^N V_{ij}^K},$$

где  $\Phi$  – признак фактора;  $N$  – число критериев качества, относящихся к  $i$ -му фактору, реально используемых при оценке.

10. Общая оценка качества в целом формируется экспертами по набору полученных значений факторов качества.

*Достоинства* метода оценки качества, основанного на иерархической модели:

1. Метод позволяет накапливать статистический материал о состоянии различных подклассов ПС в отношении значений метрик и оценочных элементов. Это создает предпосылки для определения их нормативных (базовых) значений по подклассам ПС и может служить основой для деятельности по стандартизации в области программного обеспечения.

2. Списки значений метрик и оценочных элементов являются основой для деятельности по управлению качеством в процессе разработки ПС.

3. Возможно создание инструментальных средств с целью автоматизации оценки качества ПС для тех показателей, которые такую оценку допускают.



## 9.5. Метод оценки качества программных средств по СТБ ИСО/МЭК 9126–2003

СТБ ИСО/МЭК 9126–2003 [14] определяет метод оценки качества ПС, основанный на *трехуровневой иерархической модели качества*. На первом уровне модели находятся шесть характеристик качества (см. подразд. 9.3 пособия). Второй уровень составляют подхарактеристики и третий – метрики качества.

Модель процесса оценки, положенная в основу рассматриваемого метода, приведена на рис. 50. Данная модель отражает основные стадии и этапы, требуемые для оценки качества ПС.

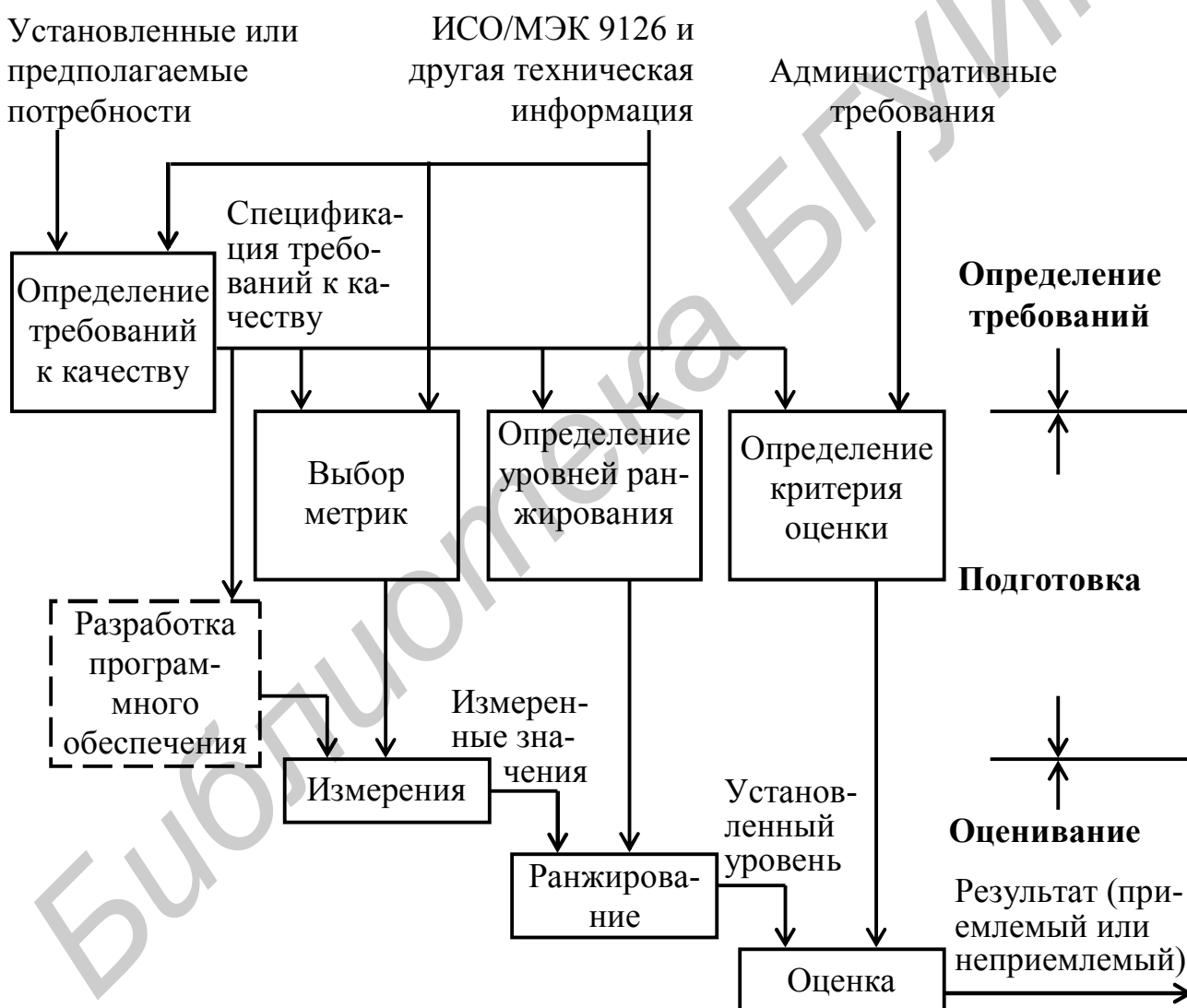


Рис. 50. Модель процесса оценки по СТБ ИСО/МЭК 9126–2003

Процесс оценки состоит из *трех стадий*: определение требований к качеству ПС, подготовка к оцениванию и процедура оценивания. Данный процесс может применяться после любой подходящей работы жизненного цикла для

каждого компонента программного продукта.

### ***Стадия 1. Определение требований к качеству***

Целью данной стадии является установка требований в терминах характеристик и подхарактеристик качества. Требования выражают потребности внешнего окружения ПС и должны быть определены до начала разработки. Так как ПС разделяется на компоненты, то требования для ПС в целом могут отличаться от требований для отдельных компонентов.

### ***Стадия 2. Подготовка к оцениванию***

Целью второй стадии является подготовка основы для оценивания. Данная стадия состоит из трех этапов.

#### ***Этап 2.1. Выбор метрик качества***

С учетом регламентированной в *СТБ ИСО/МЭК 9126–2003* иерархической модели качества уровень характеристик качества ПС определяется уровнем входящих в них подхарактеристик, а значения подхарактеристик в свою очередь определяются значениями входящих в них метрик.

В стандарте *СТБ ИСО/МЭК 9126–2003* набор рекомендуемых метрик отсутствует. Поэтому существует потребность в установлении метрик, которые соотносятся с подхарактеристиками, а следовательно, и с характеристиками ПС. Каждый количественный признак и каждое количественно оцениваемое взаимодействие ПС с его окружением, которые соотносятся с характеристикой, могут быть приняты в качестве метрики. Метрики, используемые в процессе разработки, должны быть соотнесены с соответствующими метриками пользователя, потому что метрики пользователя являются решающими.

#### ***Этап 2.2. Определение уровней ранжирования***

Для измерения количественных признаков ПС используются метрики качества. Измеренные значения отображаются на некоторой шкале. Данные значения не показывают уровень удовлетворения требований к качеству ПС. Для этой цели шкалы метрик должны быть разделены на диапазоны, соответствующие различным степеням удовлетворения требований.

В стандарте *ISO/IEC 14598–1:1999* [21] приведен пример следующих диапазонов ранжирования (рис. 51):

- разделение шкалы на две категории: неудовлетворительно и удовлетворительно;
- разделение шкалы на четыре категории (отлично, хорошо, удовлетворительно, неудовлетворительно), ограниченные соответственно запланированным уровнем, текущим уровнем для существующего или альтернативного продукта и уровнем худшего случая.

Текущий уровень определяется для управления тем, чтобы новая система не становилась хуже по сравнению с существующей. Запланированный уровень определяет уровень, который считается достижимым при доступных ресурсах. Уровень худшего случая определяет границу принятия пользователем в случае, если изделие не удовлетворяет запланированному уровню (см. рис. 51). Так как качество ПС связано с конкретными потребностями, общие уровни ранжирова-

ния невозможны и должны определяться для каждого конкретного оценивания.

### *Этап 2.3. Определение критерия оценки*

Для определения общего качества ПС должна быть учтена вся совокупность результатов оценивания различных метрик. Оценщик должен подготовить для этого процедуры, используя, например, таблицы решений или средние взвешенные значения. Обычно при этом учитываются и другие аспекты, такие как время и стоимость, которые являются косвенными факторами качества ПС.

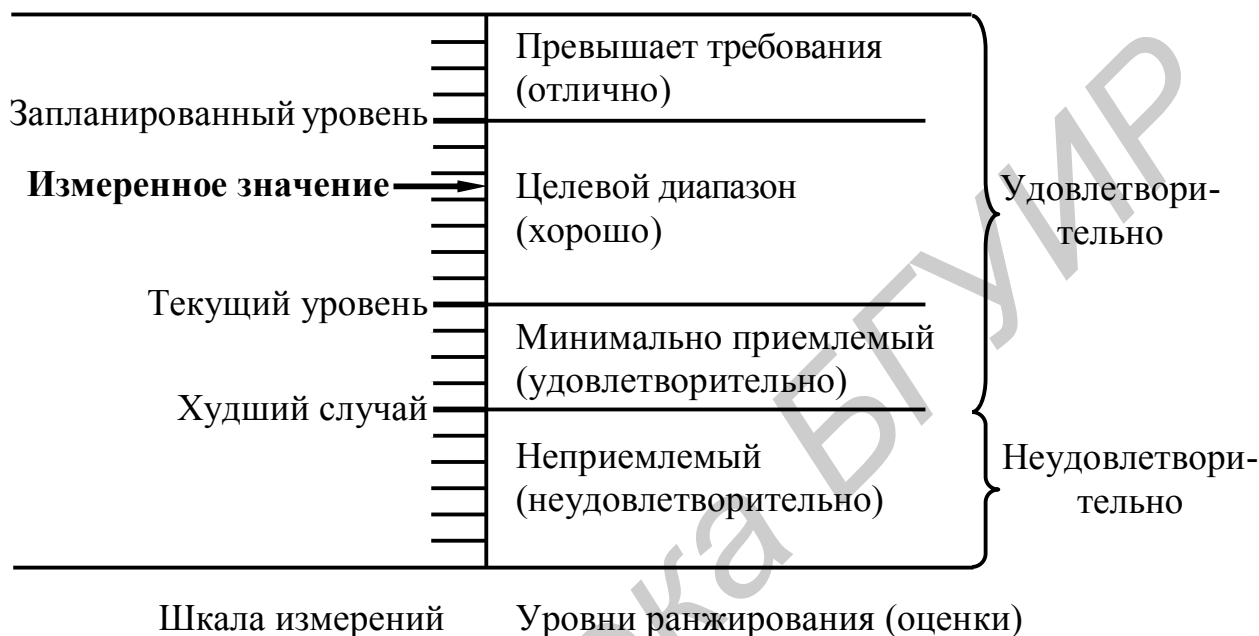


Рис. 51. Варианты ранжирования измеренных значений метрик по ISO/IEC 14598-1:1999

### *Стадия 3. Процедура оценивания*

Последняя стадия модели процесса оценивания реализуется тремя этапами: «Измерение», «Ранжирование» и «Оценка».

#### *Этап 3.1. Измерение*

Для измерения выбранные метрики применяются к ПС. Результатом являются значения в масштабах метрик.

#### *Этап 3.2. Ранжирование*

На этапе ранжирования устанавливается уровень ранжирования для измеренного значения (см. рис. 51).

#### *Этап 3.3. Оценка*

Оценка является последним этапом процесса оценивания ПС, на котором обобщается множество установленных уровней. Результатом является заключение о качестве ПС (приемлемый или неприемлемый уровень качества).

К недостаткам данного метода оценки качества следует отнести отсутствие рекомендуемых вариантов метрик и представление метода лишь в общем виде (в виде модели). Это затрудняет его конкретное использование.

# 10. СТАНДАРТИЗАЦИЯ КАЧЕСТВА ПРОГРАММНЫХ СРЕДСТВ ЗА РУБЕЖОМ

## 10.1. Стандарты серии ISO/IEC в области оценки качества программных средств

В течение десяти лет (с 1991 по 2001 г.) основой регламентирования характеристик качества ПС за рубежом являлся международный стандарт *ISO/IEC 9126:1991 – Информационная технология – Оценка программного продукта – Характеристики качества и руководства по их применению* [48]. В разд. 9 пособия описаны положения стандарта *СТБ ИСО/МЭК 9126–2003* [14], являющегося аутентичным переводом вышеназванного стандарта.

В настоящее время стандарт *ISO/IEC 9126:1991* заменен на две взаимосвязанные серии стандартов: *ISO/IEC 9126–1–4* и *ISO/IEC 14598–1–6* [49 – 52, 21 – 26].

Стандарт *ISO/IEC 9126–1–4* регламентирует иерархическую модель качества программных средств. На верхнем уровне модели находятся характеристики. Характеристики разделяются на подхарактеристики. Подхарактеристики определяются метриками. Метрики измеряют атрибуты (свойства) ПС.

Данный стандарт состоит из четырех частей под общим названием *Программная инженерия – Качество продукта* [49 – 52]:

- *ISO/IEC 9126–1:2001* – Часть 1: Модель качества;
- *ISO/IEC TR 9126–2:2003* – Часть 2: Внешние метрики;
- *ISO/IEC TR 9126–3:2003* – Часть 3: Внутренние метрики;
- *ISO/IEC TR 9126–4:2004* – Часть 4: Метрики качества в использовании.

Первая часть стандарта *ISO/IEC 9126–1:2001* по существу является пересмотренной редакцией стандарта *ISO/IEC 9126:1991*. В данной части определены два верхних уровня (характеристики и подхарактеристики) иерархической модели качества, приведены общие требования к метрикам качества, даны рекомендации по их выбору. При этом сохранена та же номенклатура из шести базовых характеристик качества ПС (см. подразд. 9.3 пособия). Однако в отличие от *ISO/IEC 9126:1991* подхарактеристики второго уровня стали нормативными, а не рекомендуемыми, определены две части модели качества (модель внутреннего и внешнего качества и модель качества в использовании) и исключен процесс оценки качества (он теперь содержится в стандарте *ISO/IEC*

14598). В данной части стандарта регламентированы следующие **виды метрик**:

- внутренние метрики;
- внешние метрики;
- метрики качества в использовании.

Вторая – четвертая части стандарта *ISO/IEC TR 9126–2–4* в настоящее время еще находятся в стадии разработки. Поэтому они опубликованы в виде технических отчетов (TR). Совокупности метрик, перечисленные в данных частях, являются рекомендуемыми, их набор не является исчерпывающим. Метрики могут модифицироваться. Возможно применение метрик, не включенных в данные части. В этих частях стандарта содержатся пояснения к применению метрик, к типам шкал метрик и типам измерений, примеры метрик для каждой подхарактеристики, примеры применения метрик на протяжении ЖЦ ПС.

Во второй части стандарта *ISO/IEC TR 9126–2:2003* определяются метрики количественного измерения внешнего качества ПС. **Внешние метрики** – это метрики, предназначенные для измерения качества программного продукта путем измерения поведения системы, частью которой является данный продукт. Внешние метрики могут использоваться в процессе эксплуатации и на стадиях тестирования или испытаний в процессах разработки и сопровождения ПС, когда уже созданы исполнимые коды программного продукта.

В третьей части стандарта *ISO/IEC TR 9126–3:2003* определяются метрики количественного измерения внутреннего качества ПС. **Внутренние метрики** – это метрики, измеряющие собственные свойства ПС. Они измеряются в процессе разработки ПС на основе спецификации требований, результатов проектирования, исходного кода или другой документации ПС. Внутренние метрики дают возможность оценить качество промежуточных программных продуктов разработки, предсказывая качество конечного программного средства.

В четвертой части стандарта *ISO/IEC TR 9126–4:2004* определяются метрики количественного измерения качества в использовании. **Метрики качества в использовании** – это метрики, измеряющие соответствие продукта потребностям заданных пользователей в достижении заданных целей с результативностью, продуктивностью, безопасностью и удовлетворением в заданных контекстах использования. Очевидно, что данные метрики могут использоваться только в процессе эксплуатации ПС в реальной среде окружения. Метрики качества в использовании основаны на измерении поведения типичных пользователей и системы, содержащей данное программное средство.

Стандарт *ISO/IEC 14598–1–6* определяет процессы оценки качества программного продукта, содержит руководство и требования к оценке. Стандарт может применяться при разработке, приобретении и независимой оценке программного средства. Данный стандарт состоит из шести частей [21 – 26]:

- *ISO/IEC 14598–1:1999* – Информационная технология – Оценка программного продукта – Часть 1: Общий обзор;
- *ISO/IEC 14598–2:2000* – Программная инженерия – Оценка продукта – Часть 2: Планирование и управление;

- *ISO/IEC 14598–3:2000* – Программная инженерия – Оценка продукта – Часть 3: Процесс для разработчиков;
- *ISO/IEC 14598–4:1999* – Программная инженерия – Оценка продукта – Часть 4: Процесс для заказчиков;
- *ISO/IEC 14598–5:1998* – Информационная технология – Оценка программного продукта – Часть 5: Процесс для оценщиков;
- *ISO/IEC 14598–6:2001* – Программная инженерия – Оценка продукта – Часть 6: Документация модулей оценки.

В первой части стандарта *ISO/IEC 14598–1:1999* приведен обзор остальных частей, определена связь *ISO/IEC 14598* со стандартами *ISO/IEC 9126–1–4* и *ISO/IEC 12207*. В данной части содержатся общие требования к спецификации и оценке качества, разъясняются концепции оценки. Устанавливаются требования к методам измерений и оценки программных продуктов. Определяется общий процесс оценки качества программного продукта (см. подразд. 10.6).

Вторая часть стандарта *ISO/IEC 14598–2:2000* содержит требования и руководство по поддержке оценки. В данной части приводятся концепции планирования и управления процессом оценки качества программного продукта, рассматривается содержание плана количественной оценки качества. Эта часть стандарта предназначена для применения на уровне организации или ее подразделений.

Третья часть стандарта *ISO/IEC 14598–3:2000* предназначена для организаций – разработчиков ПС. В ней приводятся концепции оценки и требования к процессу оценки. Данная часть ориентирована на выполнение оценки ПП, используя собственный технический персонал. Внимание в данной части сконцентрировано на оценках, которые могут предсказать качество конечного ПП на основе измерений промежуточных ПП жизненного цикла разработки.

Четвертая часть стандарта *ISO/IEC 14598–4:1999* предназначена для организаций, которые планируют приобретать готовый или разрабатываемый программный продукт. В ней определена связь работ процесса заказа из стандарта *ISO/IEC 12207–1995* с работами, выполняемыми при оценке ПП. Рассмотрены особенности выполнения оценки для имеющегося в наличии готового ПП, для приемки заказного ПП и для выбора из альтернативного числа готовых ПП.

Пятая часть стандарта *ISO/IEC 14598–5:1998* предназначена для использования оценщиком, выполняющим независимую оценку программного продукта. Как правило, персонал оценки работает в независимой организации. В данной части приводятся концепции оценки и требования к процессу оценки. Рассмотрена структура отчета об оценке. Приводятся рекомендации по выбору уровней ранжирования при проведении измерений. Оценка качества программного продукта оценщиком может выполняться по запросу разработчика, заказчика (покупателя) или другой стороны.

Шестая часть стандарта *ISO/IEC 14598–6:2001* предназначена для поддержки оценки программного продукта и содержит руководство по документи-

рованию модулей оценки. **Модуль оценки** представляет собой полностью укомплектованную информацию, необходимую для проведения процесса оценки некоторой характеристики или подхарактеристики качества. Модуль содержит спецификацию соответствующей модели качества (характеристика, подхарактеристики, внутренние или внешние метрики качества), методики и процедуры оценки, входные данные, связанные с оценкой, информацию о запланированном применении модели и о ее фактическом применении, структуру типового отчета о результатах выполненной оценки. Рассмотрен ряд примеров модулей оценки. Для каждого процесса оценки должны выбираться соответствующие модули оценки. Данная часть стандарта может быть использована организациями, разрабатывающими новые модули оценки и производящими оценку ПС.

## **10.2. Связь качества программного средства с его жизненным циклом**

В течение жизненного цикла программного средства его качество изменяется. Требуемое качество, определенное в начале ЖЦ, отличается от фактического качества поставленного продукта. Существует несколько точек зрения на качество ПС в течение его ЖЦ. От их выбора зависит как оценка качества ПС, так и управление качеством на каждой стадии жизненного цикла.

С учетом этого для различных стадий ЖЦ стандарт *ISO/IEC 9126-1:2001* определяет следующие **виды качества программных средств** [49]:

- *потребности пользователя в качестве* определяются как требования к качеству, выраженные в терминах метрик качества в использовании, внешних и иногда внутренних метрик; эти требования должны применяться как критерии при аттестации продукта; получение ПП, удовлетворяющего потребностям пользователя, обычно требует итеративного подхода к разработке программного средства с постоянной обратной связью с потенциальным пользователем;
- *требования к внешнему качеству* определяют требуемый уровень качества с внешней точки зрения; они включают требования, вытекающие из потребностей пользователя в качестве, включая требования к качеству в использовании; требования к внешнему качеству применяются как цель при аттестации продукта на различных стадиях разработки; данные требования для всех характеристик качества, определенных в *ISO/IEC 9126-1:2001*, во-первых, должны быть выражены в спецификации требований к качеству, используя внешние метрики, во-вторых, должны быть преобразованы в требования к внутреннему качеству и, в-третьих, должны использоваться как критерии при оценке продукта;
- *требования к внутреннему качеству* определяют требуемый уровень качества с внутренней точки зрения на программный продукт; они используются для определения свойств промежуточных продуктов разработки; промежуточные продукты могут включать статические и динамические модели, другие

документы и исходный код ПП; требования к внутреннему качеству могут использоваться как цель при аттестации продукта на различных стадиях разработки; они могут использоваться для определения стратегий разработки и критериев оценки и верификации в течение разработки; требования к внутреннему качеству должны определяться количественно, используя внутренние метрики;

- *внутреннее качество* – совокупность характеристик программного продукта с внутренней точки зрения; внутреннее качество измеряется с помощью внутренних метрик и оценивается по отношению к требованиям к внутреннему качеству; отдельные элементы качества ПП могут улучшаться при реализации кода, проверке или тестировании, но фундаментальная основа качества программного продукта, представленная внутренним качеством, остается неизменной до повторного проектирования;

- *оценочное (или прогнозируемое) внешнее качество* – оцененное или предсказанное качество конечного программного продукта на каждой стадии процесса разработки для каждой характеристики качества, основанное на знании внутреннего качества;

- *внешнее качество* – совокупность характеристик программного продукта с внешней точки зрения; это качество, измеряемое и оцениваемое на основе внешних метрик при выполнении ПП во время тестирования (испытаний) в моделируемой среде с моделируемыми данными или во время эксплуатации;

- *оценочное (или прогнозируемое) качество в использовании* – оцененное или предсказанное качество конечного программного продукта на каждой стадии процесса разработки для каждой характеристики качества в использовании, основанное на знании внутреннего и внешнего качества;

- *качество в использовании* – качество программного продукта, применяемого в заданной среде и заданном контексте использования, с точки зрения пользователя; оно оценивается на основе метрик качества в использовании и в первую очередь измеряет степень достижения пользователем своих целей в конкретной среде, а не свойства самого ПП; пользователь оценивает только те атрибуты программного продукта, которые он применяет в своих задачах.

Рис. 52 иллюстрирует изменение и взаимосвязь различных видов качества в жизненном цикле программных средств [21].

### **10.3. Модель внешнего и внутреннего качества ПС**

Как уже было отмечено, в стандарте *ISO/IEC 9126-1:2001* регламентированы две части модели качества программных средств: модель внутреннего и внешнего качества и модель качества в использовании. Данные модели различаются в зависимости от представления качества в ЖЦ ПС (см. рис. 52). Эти модели могут быть использованы, например, в следующих случаях:

- проверка полноты определения требований;



- определение требований к ПС;
- определение целей проектирования ПС;
- определение целей испытаний ПС;
- определение критериев обеспечения качества;
- определение критериев приемки заверченного ПС.

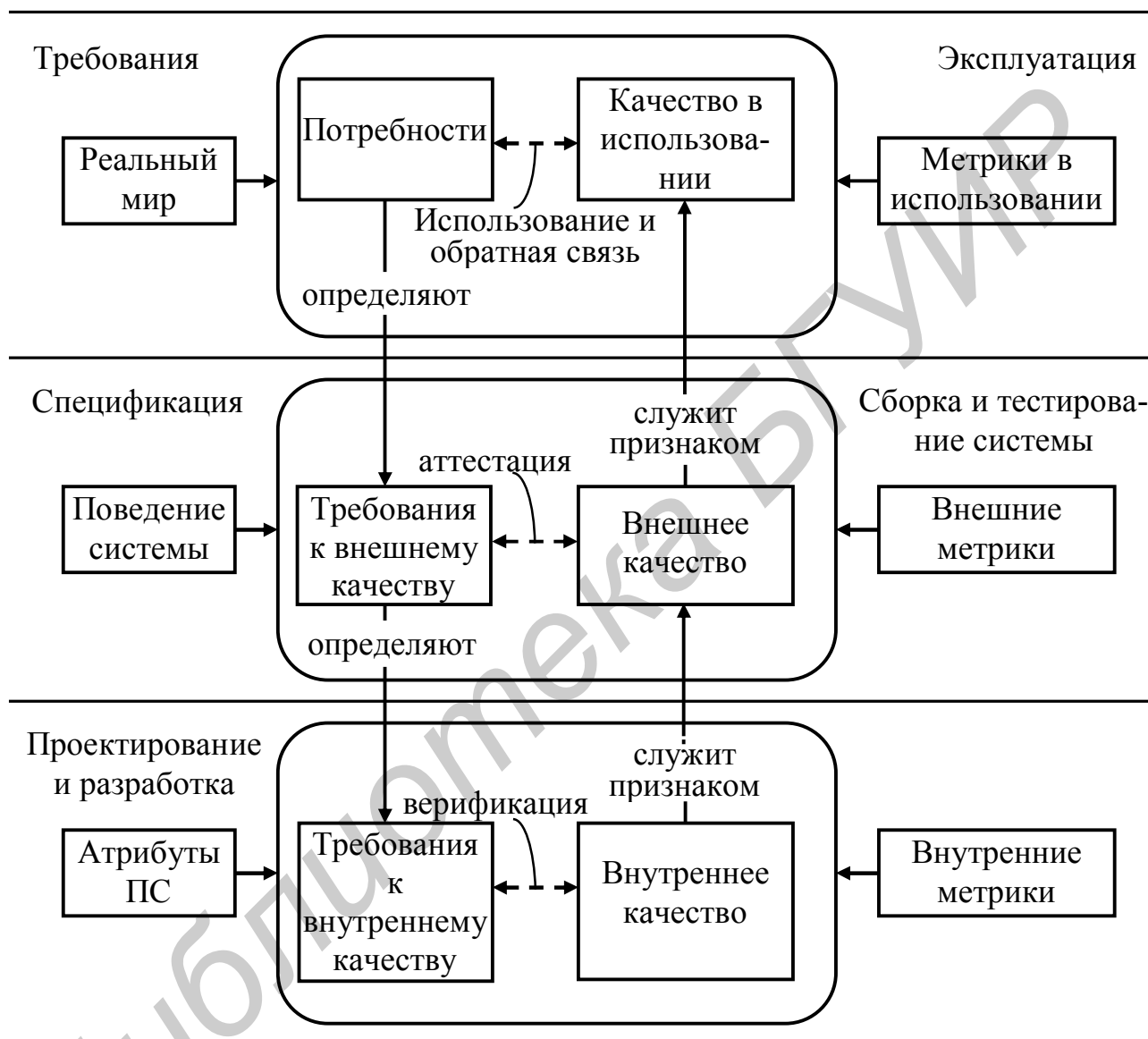


Рис. 52. Качество в жизненном цикле программных средств по ISO/IEC 14598–1:1999

На рис. 53 приведены два верхних уровня модели внешнего и внутреннего качества. На верхнем уровне находится **шесть основных характеристик качества программных средств**. Это *функциональность, надежность, практичность, эффективность, сопровождаемость и мобильность*.

Подхарактеристики, находящиеся на втором уровне, также стандартизованы. Они могут измеряться внутренними или внешними метриками.

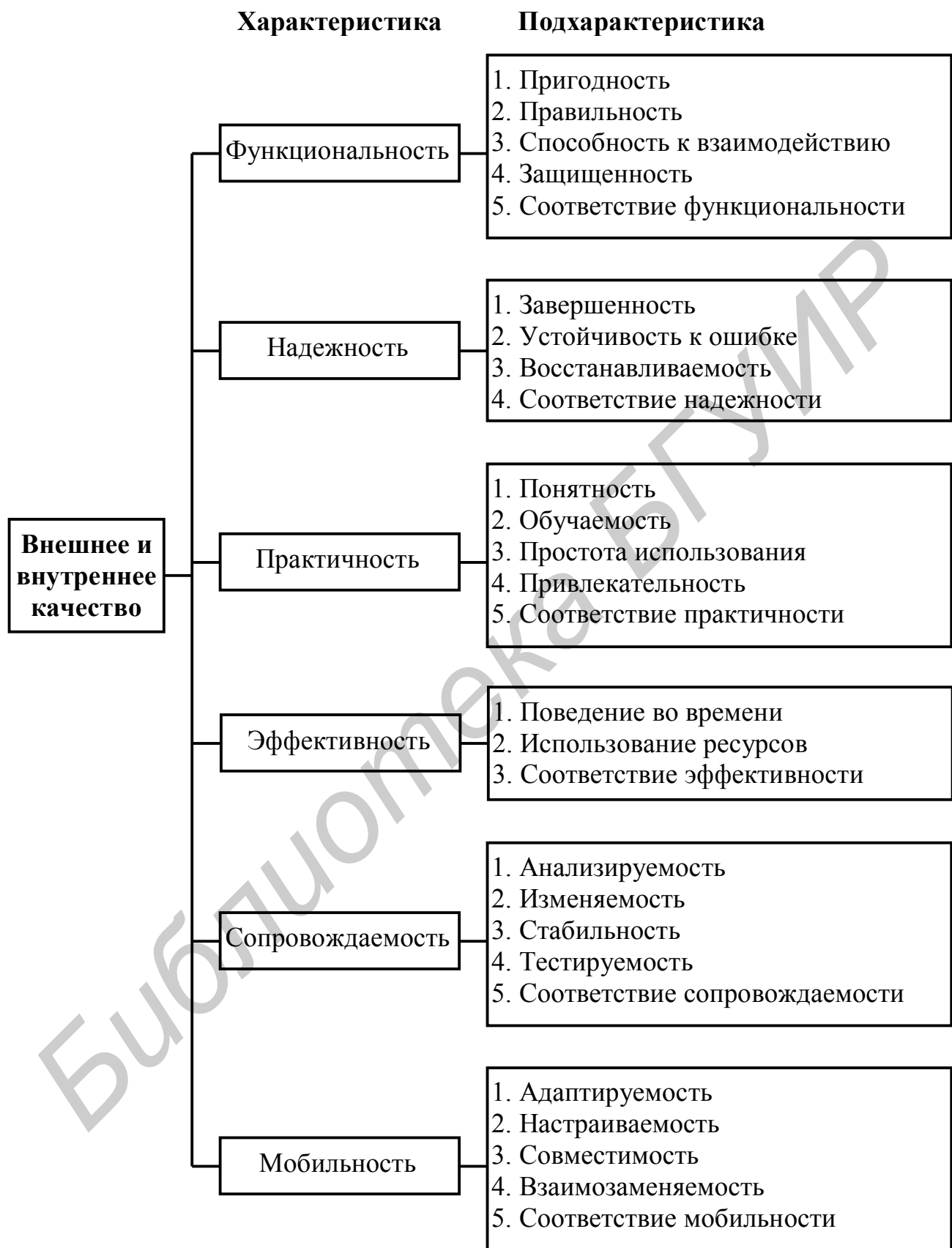


Рис. 53. Модель внешнего и внутреннего качества по ISO/IEC 9126–1:2001

### 10.3.1. Функциональность

**Функциональность (Functionality)** – способность программного продукта обеспечивать функции, удовлетворяющие установленные и подразумеваемые потребности при применении программного средства в заданных условиях. Эта характеристика определяет, *что* делает ПС в соответствии с потребностями. Другие характеристики определяют, *когда* и *как* эти потребности удовлетворяются. Подхарактеристики *Функциональности* приведены на рис. 53.

**Пригодность (Suitability)** – способность программного продукта обеспечивать набор функций, соответствующий специфическим задачам и целям пользователей.

**Правильность (корректность, Accuracy)** – способность программного продукта обеспечивать правильные или приемлемые результаты или эффекты с необходимой степенью точности.

**Способность к взаимодействию (Interoperability)** – способность программного продукта взаимодействовать с одной или несколькими заданными системами.

**Защищенность (Security)** – способность программного продукта защищать информацию и данные так, чтобы несанкционированные пользователи или системы не могли прочесть или модифицировать их, а санкционированные пользователи или системы не могли получить отказа в доступе к ним.

**Соответствие функциональности (Functionality compliance)** – способность программного продукта соответствовать стандартам, соглашениям или нормам законов, связанным с функциональностью.

### 10.3.2. Надежность

**Надежность (Reliability)** – способность программного продукта поддерживать заданный уровень качества функционирования при его использовании в заданных условиях. Ограничения надежности в процессе эксплуатации вызваны ошибками в требованиях, проектировании и кодировании.

Подхарактеристики *Надежности* приведены на рис. 53.

**Завершенность (Maturity)** – способность программного продукта избегать отказов вследствие ошибок в программах.

**Устойчивость к ошибке (Fault tolerance)** – способность программного продукта поддерживать заданный уровень качества функционирования в случаях ошибок в программах или нарушения заданного интерфейса ПП.

**Восстанавливаемость (Recoverability)** – способность программного продукта восстанавливать заданный уровень качества функционирования и данные, поврежденные в случае отказа. Одним из показателей восстанавливаемости является длительность восстановления.

**Готовность (доступность, Availability)** – способность программного продукта быть в состоянии выполнять требуемую функцию в данный момент

времени при заданных условиях использования. Готовность может оцениваться отношением времени, в течение которого ПС находится в работоспособном состоянии, к общему времени применения. Следовательно, готовность – это комбинация завершенности (которая влияет на частоту отказов), устойчивости к ошибке и восстанавливаемости (которая влияет на продолжительность восстановления). По этой причине готовность не включена в модель внутреннего и внешнего качества в виде отдельной подхарактеристики (см. рис. 53).

**Соответствие надежности (Reliability compliance)** – свойство программного продукта соответствовать стандартам, соглашениям и нормативным документам, связанным с надежностью.

### 10.3.3. Практичность

**Практичность (Usability)** – способность программного продукта быть понятным, изученным, использованным и привлекательным для пользователя при применении в заданных условиях. Очевидно, что практичность зависит от некоторых аспектов таких характеристик качества, как функциональность, надежность и эффективность. Подхарактеристики *Практичности* приведены на рис. 53.

**Понятность (Understandability)** – способность программного продукта, обеспечивающая понимание пользователем пригодности и способа использования программного средства для конкретных задач и условий применения.

**Обучаемость (Learnability)** – способность программного продукта, обеспечивающая изучение пользователем принципов его применения.

**Простота использования (Operability)** – способность программного продукта, позволяющая пользователю эксплуатировать его и управлять им. На простоту использования влияют некоторые аспекты таких подхарактеристик, как пригодность, изменяемость, адаптируемость и простота внедрения. Простота использования зависит, например, от контролируемости и устойчивости к ошибке.

**Привлекательность (Attractiveness)** – способность программного продукта нравиться пользователю. Данная подхарактеристика связана со свойствами оформления ПП (например использование цветов, графики и т.п.).

**Соответствие практичности (Usability compliance)** – свойство программного продукта соответствовать стандартам, соглашениям и руководствам, связанным с практичностью.

### 10.3.4. Эффективность

**Эффективность (Efficiency)** – способность программного продукта обеспечить соответствующую производительность в зависимости от количества используемых вычислительных ресурсов в заданных условиях. Ресурсы могут включать другие программные продукты, конфигурацию программных и аппаратных средств системы и материалы.

Подхарактеристики *Эффективности* приведены на рис. 53.

**Поведение во времени (Time behaviour)** – способность программного продукта обеспечивать соответствующие времена отклика и обработки, а также пропускную способность при выполнении своих функций в заданных условиях.

**Использование ресурсов (Resource utilisation)** – способность программного продукта использовать соответствующее количество всех типов ресурсов при выполнении своих функций в заданных условиях. В данной подхарактеристике человеческие ресурсы не учитываются. Они учитываются в такой характеристике качества в использовании, как продуктивность (см. подразд. 10.4).

**Соответствие эффективности (Efficiency compliance)** – способность программного продукта соответствовать стандартам и соглашениям, связанным с эффективностью.

### 10.3.5. Сопровождаемость

**Сопровождаемость (Maintainability)** – способность программного продукта к модификации. Модификации могут включать исправления, усовершенствования или адаптацию ПС к изменениям в среде применения, в требованиях и функциональных спецификациях. Подхарактеристики *Сопровождаемости* представлены на рис. 53.

**Анализируемость (Analysability)** – способность программного продукта к диагностике его недостатков или причин отказов или к идентификации его частей, которые должны быть модифицированы.

**Изменяемость (Changeability)** – способность программного продукта к реализации заданной модификации. Реализация включает проектирование, кодирование и изменение документации.

**Стабильность (Stability)** – способность программного средства предотвращать непредвиденные эффекты от его модификации.

**Тестируемость (Testability)** – способность программного продукта к проверке результата модификации.

**Соответствие сопровождаемости (Maintainability compliance)** – способность программного продукта соответствовать стандартам или соглашениям, связанным с сопровождением.

### 10.3.6. Мобильность

**Мобильность (Portability)** – способность программного продукта к переносу из одной среды в другую. Среда может включать организационное, аппаратное и программное окружение. Подхарактеристики *Мобильности* представлены на рис. 53.

**Адаптируемость (Adaptability)** – способность программного продукта к адаптации к различным окружающим средам без применения дополнительных действий или средств. Адаптируемость включает масштабируемость внутренних возможностей (например, областей экранов, таблиц, форматов отчетов).

**Настраиваемость (простота внедрения, *Installability*)** – способность программного продукта устанавливаться в заданной среде окружения.

**Совместимость (*Co-existence*)** – способность программного продукта к сосуществованию с другими независимыми программными средствами в общей среде, разделяя общие ресурсы.

**Взаимозаменяемость (*Replaceability*)** – способность программного продукта к использованию вместо другого (заданного) ПП с той же целью и в той же среде. Например, для пользователя важна взаимозаменяемость новой версии ПП с его старой версией. Взаимозаменяемость может включать атрибуты таких подхарактеристик, как настраиваемость и адаптируемость.

**Соответствие мобильности (*Portability compliance*)** – способность программного продукта соответствовать стандартам или соглашениям, связанным с мобильностью.

## 10.4. Модель качества в использовании

Второй частью модели качества, определенной в стандарте *ISO/IEC 9126-1:2001* [49], является модель качества в использовании. Достижение качества в использовании зависит от достижения необходимого внешнего качества, которое, в свою очередь, зависит от достижения необходимого внутреннего качества (см. рис. 52).

**Качество в использовании (*Quality in use*)** – это способность программного продукта позволять заданным пользователям достигать заданные цели с результативностью, продуктивностью, безопасностью и удовлетворением в заданном контексте использования.

Качество в использовании – это восприятие пользователем качества окружающей среды, содержащей ПП. Оно измеряется больше на основе результатов использования ПП в среде, чем на основе собственных внутренних свойств ПП.

Модель качества в использовании является иерархической. На ее верхнем уровне находятся четыре характеристики. Характеристики определяются непосредственно метриками (второй уровень модели). В отличие от модели внешнего и внутреннего качества (см. рис. 53) уровень подхарактеристик в модели качества в использовании отсутствует.

На рис. 54 приведен верхний уровень (уровень характеристик) модели качества в использовании. В соответствии с данной моделью качество в использовании разделяется на четыре характеристики: *результативность, продуктивность, безопасность, удовлетворенность*.

**Результативность (*Effectiveness*)** – это способность программного продукта, позволяющая пользователям достигать заданные цели с точностью и полнотой в заданном контексте использования.

**Продуктивность (*Productivity*)** – это способность программного продукта, позволяющая пользователям расходовать количество ресурсов, соответствующее результативности, достигаемой в заданном контексте использования. Ре-

сурсы могут включать время выполнения задачи, усилия пользователя, материалы, стоимость использования.

**Безопасность (Safety)** – это способность программного продукта достигать приемлемых уровней риска причинения вреда людям, бизнесу, программному обеспечению, имуществу или окружающей среде в заданном контексте использования. Обычно *риски* – это результат дефектов в функциональности (включая защищенность), надежности, практичности и сопровождаемости.

**Удовлетворенность (Satisfaction)** – это способность программного продукта удовлетворять пользователя в заданном контексте использования. Удовлетворенность определяется реакцией пользователя на взаимодействие с программным продуктом и включает отношение к применению продукта.

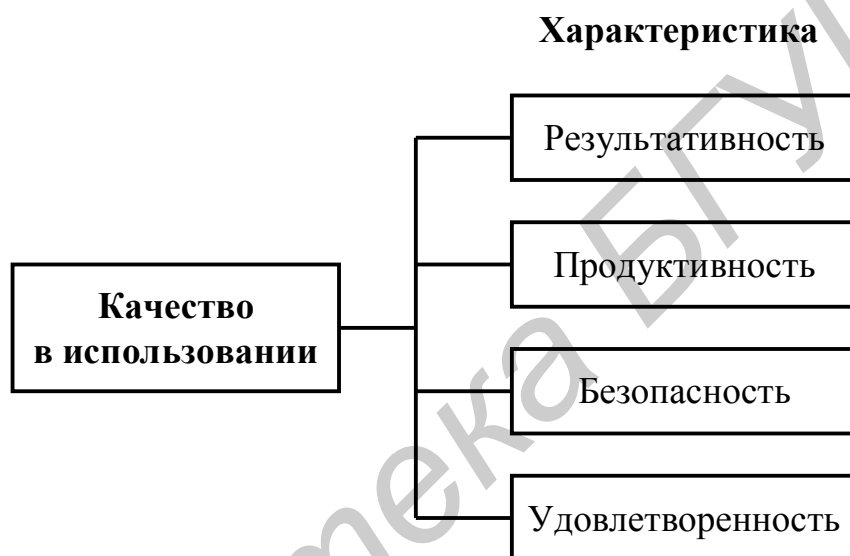


Рис. 54. Модель качества в использовании

## 10.5. Метрики качества программных средств

### 10.5.1. Свойства и критерии обоснованности метрик

Как было описано выше (см. подразд. 10.1), стандарт *ISO/IEC 9126–1:2001* классифицирует метрики качества ПС на внутренние, внешние и метрики качества в использовании. В модели внешнего и внутреннего качества метрики находятся на третьем уровне иерархии и определяют значения подхарактеристик качества. В модели качества в использовании метрики находятся на втором уровне иерархии и непосредственно определяют значения характеристик каче-

ства. Применение конкретного вида метрик определяется стадией жизненного цикла программного средства (см. подразд. 10.2).

Вторая, третья и четвертая части стандарта *ISO/IEC TR 9126-2-4* [50 – 52] посвящены детальному рассмотрению соответственно внешних и внутренних метрик качества программных средств и метрик качества в использовании.

В *Приложении А* данных частей стандарта определены следующие **желательные свойства метрик**:

1) *надежность*; надежность связана со случайной ошибкой; метрика свободна от случайной ошибки, если случайные изменения не влияют на результаты метрики;

2) *повторяемость*; повторное использование метрики для того же продукта теми же специалистами по оценке, используя ту же спецификацию оценки (включая ту же окружающую среду), тот же тип пользователей и окружения, должно привести к тем же результатам с соответствующими допусками; соответствующие допуски должны учитывать такие компоненты, как усталость и результат накопленных познаний;

3) *однотипность*; применение метрики для того же продукта различными специалистами по оценке, используя ту же спецификацию оценки (включая ту же окружающую среду), тот же тип пользователей и окружения, должно привести к тем же результатам с соответствующими допусками;

4) *применимость*; метрика должна четко указывать условия (например, наличие определенных атрибутов), которые ограничивают её употребление;

5) *показательность*; это способность метрики идентифицировать части или элементы программы, которые должны быть улучшены, на основании сравнения измеренных и ожидаемых результатов;

6) *корректность*; метрика должна обладать следующими свойствами:

- *объективность*; результаты метрики и её входные данные должны быть основаны на фактах и не подвластны чувствам или мнениям специалистов по оценке или тестированию (исключая метрики удовлетворенности или привлекательности, с помощью которых измеряются чувства и мнения пользователя);

- *беспристрастность*; измерение не должно быть направлено на получение какого-либо специфического результата;

- *адекватность точности*; точность определяется при проектировании метрики и особенно при выборе описаний фактов, используемых как основа для метрики; разработчик метрики должен описать точность и чувствительность метрики;

7) *значимость*; измерение должно давать значащие результаты, касающиеся поведения программы или характеристик качества.

Метрика должна также быть *эффективной по отношению к стоимости*. Это значит, что более дорогие метрики должны обеспечивать лучшие результаты оценки.

Разработчик метрики должен доказать ее обоснованность. Метрика должна



удовлетворять хотя бы одному из следующих **критериев обоснованности метрики**:

1) *корреляция*; изменение в значениях характеристик качества (оперативно определенных по результатам измерения основных метрик), обусловленное изменением в значениях метрики, должно определяться линейной зависимостью;

2) *трассировка*; если метрика  $M$  непосредственно связана с величиной характеристики качества  $Q$  (оперативно определенной по результатам измерения основных метрик), то изменение величины  $Q$  ( $T1$ ), имеющейся в момент времени  $T1$ , к величине  $Q$  ( $T2$ ), полученной в момент времени  $T2$ , должно сопровождаться изменением значения метрики от  $M$  ( $T1$ ) до  $M$  ( $T2$ ) в том же направлении (например, если увеличивается  $Q$ , то  $M$  тоже увеличивается);

3) *непротиворечивость*; если значения характеристик качества (оперативно полученные по результатам измерения основных метрик)  $Q1, Q2, \dots, Qn$ , связанные с продуктами или процессами  $1, 2, \dots, n$ , определяются соотношением  $Q1 > Q2 > \dots > Qn$ , то соответствующие значения метрики должны удовлетворять соотношению  $M1 > M2 > \dots > Mn$ .

4) *предсказуемость*; если метрика используется в момент времени  $T1$  для прогноза значения (оперативно полученного по результатам измерения основных метрик) характеристики качества  $Q$  в момент времени  $T2$ , то ошибка прогнозирования, определяемая выражением

$$\frac{\text{прогнозное } Q(T2) - \text{фактическое } Q(T2)}{\text{фактическое } Q(T2)},$$

должна попадать в допустимый диапазон ошибок прогнозирования;

5) *селективность*; метрика должна быть способной различать высокое и низкое качество программного средства.

В стандартах *ISO/IEC 9126-2, -3, -4* для каждой подхарактеристики внешнего и внутреннего качества и характеристики качества в использовании приведены таблицы, в которых даны примеры метрик качества.

*Таблицы* имеют следующую структуру:

- 1) название метрики;
- 2) назначение метрики (изложено в виде вопроса, на который отвечает применение метрики);
- 3) метод применения;
- 4) способ измерения, формула, исходные и вычисляемые данные;
- 5) интерпретация измеренного значения (диапазон и предпочтительные значения);
- 6) тип шкалы, используемой при измерении метрики (номинальная, порядковая, интервальная, относительная или абсолютная);
- 7) тип измеренного значения; используются следующие **типы измеренных значений**:

- *тип размера* (например, функциональный размер, размер исходного текста);

- *тип времени* (например, затраченное время, необходимое пользователю время);

- *тип количества* (например, количество изменений, количество отказов);

8) источники входных данных для измерения;

9) ссылка на *ISO/IEC 12207:1995* (процессы жизненного цикла программных средств, при выполнении которых применима метрика);

10) целевая аудитория (данное понятие приведено в подразд. 6.2 пособия).

Для обеспечения возможности совместного использования различных метрик (независимо от их физического смысла, единиц измерения и диапазонов значений) при количественной оценке качества программных продуктов метрики в стандартах *ISO/IEC TR 9126-2-4* по возможности представляются в относительных единицах в виде

$$X = A/B \quad (1)$$

или

$$X = 1 - A/B, \quad (2)$$

где  $X$  – значение метрики;  $A$  – абсолютное (измеренное) значение некоторого свойства (атрибута) оцениваемого продукта или документации;  $B$  – базовое значение соответствующего свойства.

Из двух вышеназванных формул для конкретной метрики выбирается та, которая соответствует *критериям трассировки и непротиворечивости*: с увеличением относительного значения метрики значение подхарактеристики и характеристики качества должно увеличиваться.

Вычисление метрик по формуле (1) или (2) позволяет привести их относительные значения в диапазон

$$0 \leq X \leq 1, \quad (3)$$

что упрощает их совместное использование при интегральной оценке качества программных средств.

В пп. 10.5.2 – 10.5.4 приведены примеры метрик (по одной на каждую подхарактеристику или характеристику качества) из рекомендуемых в стандартах *ISO/IEC TR 9126-2,-3,-4* [50 – 52].

Следует отметить, что не все метрики, приведенные в стандартах *ISO/IEC TR 9126-2,-3,-4*, удовлетворяют вышеприведенным свойствам, критериям, оцениваются с помощью выражений (1), (2) или попадают в диапазон (3).

### **10.5.2. Внутренние метрики качества программных средств**

*Внутренние метрики функциональности* предназначены для предсказания того, удовлетворяет ли разрабатываемый программный продукт требованиям к функциональности и предполагаемым потребностям пользователя.

*Внутренние метрики надежности* используются во время разработки

программного продукта для предсказания того, удовлетворяет ли ПП заявленным потребностям в надежности.

**Внутренние метрики практичности** используются во время разработки программного продукта для предсказания степени, в которой ПП может быть понят, изучен, управляем, привлекателен и соответствует договоренностям и руководствам по практичности.

**Внутренние метрики эффективности** используются во время разработки программного продукта для предсказания эффективности поведения ПП во время тестирования или эксплуатации.

**Внутренние метрики сопровождаемости** используются для предсказания уровня усилий, необходимых для модификации программного продукта.

**Внутренние метрики мобильности** используются для предсказания воздействия программного продукта на поведение исполнителя или системы при проведении работ по переносу.

Табл. 37 содержит примеры внутренних метрик качества ПС, содержащихся в стандарте *ISO/IEC TR 9126–3:2003* [51]. Во втором столбце таблицы по каждой подхарактеристике приведено название одной метрики, уникальная формула или номер формулы (1) или (2) из подразд. 10.5.1 для оценки данной метрики. Исходные данные в третьем столбце – это данные, используемые в соответствующей формуле для вычисления значения представленной метрики.

Таблица 37

Внутренние метрики качества программных средств

Название подхарактеристики	Название метрики, формула для ее оценки	Исходные данные для вычисления метрики по соответствующей формуле
<b>Функциональность</b>		
Пригодность	Полнота функциональной реализации (2)	<i>A</i> – число нереализованных функций, обнаруженных при оценке; <i>B</i> – число функций, описанных в спецификации требований
Правильность	Точность (1)	<i>A</i> – количество элементов данных, реализованных с заданными уровнями точности, подтвержденное при оценке; <i>B</i> – количество элементов данных, для которых в спецификации заданы уровни точности
Способность к взаимодействию	Соответствие интерфейсов (протоколов) (1)	<i>A</i> – количество интерфейсных протоколов, реализующих заданный в спецификации формат, подтвержденных при проверке; <i>B</i> – количество интерфейсных протоколов, которые должны быть реализованы в соответствии со спецификацией

Название подхарактеристики	Название метрики, формула для ее оценки	Исходные данные для вычисления метрики по соответствующей формуле
Защищенность	Предотвращение разрушения данных (1)	<i>A</i> – количество реализованных случаев предотвращения разрушения данных из заданных в спецификации, подтвержденное при проверке; <i>B</i> – количество случаев обработки (доступа), которые определены в спецификации как способные разрушить данные
Соответствие функциональности	Соответствие функциональности (1)	<i>A</i> – количество корректно реализованных элементов, связанных с соответствием функциональности, подтвержденное при оценке; <i>B</i> – общее количество элементов соответствия
<b>Надежность</b>		
Завершенность	Полнота тестирования (1)	<i>A</i> – количество тестовых комбинаций, спроектированных в плане тестирования и подтвержденных при проверке; <i>B</i> – количество требуемых тестовых комбинаций
Устойчивость к ошибке	Предотвращение некорректных действий (1)	<i>A</i> – количество функций, реализованных с предотвращением некорректных действий; <i>B</i> – количество типичных некорректных действий, которое должно быть учтено
Восстанавливаемость	Способность к восстановлению (1)	<i>A</i> – количество реализованных требований к восстановлению, подтвержденное при проверке; <i>B</i> – общее количество требований к восстановлению, определенных в спецификации
Соответствие надежности	Соответствие надежности (1)	<i>A</i> – количество корректно реализованных элементов, связанных с соответствием надежности, подтвержденное при оценке; <i>B</i> – общее количество элементов соответствия
<b>Практичность</b>		
Понятность	Способность к демонстрации (1)	<i>A</i> – количество демонстрируемых функций, подтвержденное при проверке; <i>B</i> – общее количество функций, которые должны обладать способностью к демонстрации
Обучаемость	Полнота документации пользователя и/или возможности	<i>A</i> – количество описанных функций; <i>B</i> – общее количество предоставляемых функций

Название подхарактеристики	Название метрики, формула для ее оценки	Исходные данные для вычисления метрики по соответствующей формуле
	электронной справки help (1)	
Простота использования	Отменяемость действий пользователя (1)	$A$ – количество реализованных функций, которые могут быть отменены пользователем с восстановлением предыдущих данных; $B$ – общее количество функций
Привлекательность	Настраиваемость вида интерфейса пользователя (1)	$A$ – количество типов элементов интерфейса, которые могут быть настроены; $B$ – общее количество типов элементов интерфейса
Соответствие практичности	Соответствие практичности (1)	$A$ – количество корректно реализованных элементов, связанных с соответствием практичности, подтвержденное при оценке; $B$ – общее количество элементов соответствия
<b>Эффективность</b>		
Поведение во времени	Пропускная способность $X = A$	$A$ – число задач в единицу времени, подтвержденное при проверке
Использование ресурсов	Использование памяти $X = A$	$A$ – размер памяти в байтах (вычисленный или моделированный)
Соответствие эффективности	Соответствие эффективности (1)	$A$ – количество корректно реализованных элементов, связанных с соответствием эффективности, подтвержденное при проверке; $B$ – общее количество элементов соответствия
<b>Сопровождаемость</b>		
Анализируемость	Готовность диагностических функций (1)	$A$ – количество реализованных диагностических функций из заданных в спецификации, подтвержденное при проверке; $B$ – требуемое количество диагностических функций
Изменяемость	Регистрируемость изменений (1)	$A$ – количество изменений в функциях/модулях, отраженных в комментариях, подтвержденное при проверке; $B$ – общее количество изменений в функциях/модулях относительно оригинального кода
Стабильность	Влияние	$A$ – количество обнаруженных вредных

Название подхарактеристики	Название метрики, формула для ее оценки	Исходные данные для вычисления метрики по соответствующей формуле
	изменений (2)	влияний после модификаций; <b><i>B</i></b> – количество сделанных модификаций
Тестируемость	Полнота встроенных функций тестирования (1)	<b><i>A</i></b> – количество реализованных встроенных функций тестирования из заданных в спецификации, подтвержденное при проверке; <b><i>B</i></b> – требуемое количество встроенных функций тестирования
Соответствие сопровождаемости	Соответствие сопровождаемости (1)	<b><i>A</i></b> – количество корректно реализованных элементов, связанных с соответствием сопровождаемости, подтвержденное при оценке; <b><i>B</i></b> – общее количество элементов соответствия
<b><i>Мобильность</i></b>		
Адаптируемость	Адаптируемость структур данных (1)	<b><i>A</i></b> – количество структур данных, работоспособность которых не нарушена после адаптации, подтвержденное при проверке; <b><i>B</i></b> – общее количество структур данных, которые должны обладать способностью к адаптации
Настраиваемость	Объем работ по установке (1)	<b><i>A</i></b> – количество автоматических шагов установки (инсталляции), подтвержденное при проверке; <b><i>B</i></b> – требуемое количество шагов инсталляции
Совместимость	Доступная совместимость (1)	<b><i>A</i></b> – количество объектов, с которыми продукт может сосуществовать, из заданных в спецификации; <b><i>B</i></b> – количество объектов в окружающей среде, с которыми продукт должен сосуществовать
Взаимозаменяемость	Преемственность данных (1)	<b><i>A</i></b> – количество элементов данных ПС, которые продолжают использоваться после замещения (из заданных в спецификации), подтвержденное при проверке; <b><i>B</i></b> – количество элементов старых данных, которые должны использоваться из старого ПС
Соответствие мобильности	Соответствие мобильности (1)	<b><i>A</i></b> – количество корректно реализованных элементов, связанных с соответствием мобильности, подтвержденное при проверке; <b><i>B</i></b> – общее количество элементов соответствия

Следует обратить внимание, что приведенные в данной таблице метрики эффективности измеряются не в относительных единицах, а в абсолютных. Это затрудняет их использование при интегральной оценке качества программных продуктов. Кроме того, метрика эффективности «Использование памяти» не удовлетворяет таким критериям обоснованности метрик, как корреляция, трассировка и непротиворечивость.

### 10.5.3. Внешние метрики качества программных средств

**Внешние метрики функциональности** должны измерять свойства (атрибуты) функционального поведения системы, содержащей ПС.

**Внешние метрики надежности** должны измерять свойства, связанные с поведением системы, содержащей ПС, во время тестирования, чтобы показать степень надежности ПС в системе в процессе эксплуатации.

**Внешние метрики практичности** показывают, в какой мере программное средство может быть понято, изучено, управляемо, привлекательно и соответствует договоренностям и руководствам по практичности.

**Внешние метрики эффективности** должны измерять такие атрибуты, как характер изменения затрат времени и использования ресурсов компьютерной системы, включающей ПС, во время тестирования или эксплуатации.

**Внешние метрики сопровождаемости** измеряют такие атрибуты, как поведение персонала сопровождения, пользователя или системы, включающей ПС, при модификации ПС во время тестирования или сопровождения.

**Внешние метрики мобильности** измеряют такие атрибуты, как поведение оператора или системы при проведении работ по переносу.

Табл. 38 содержит примеры внешних метрик качества программных средств из стандарта *ISO/IEC TR 9126–2:2003* [50]. Во втором столбце таблицы по каждой подхарактеристике приведено название одной метрики, уникальная формула или номер формулы (1) или (2) из подразд. 10.5.1 для оценки данной метрики. Исходные данные в третьем столбце – это данные, используемые в соответствующей формуле для вычисления значения представленной метрики.

Таблица 38

Внешние метрики качества программных средств

Название подхарактеристики	Название метрики, формула для ее оценки	Исходные данные для вычисления метрики по соответствующей формуле
<b>Функциональность</b>		
Пригодность	Полнота функциональной	<i>A</i> – количество отсутствующих функций, обнаруженных при проверке; <i>B</i> – количество функ-

Название подхарактеристики	Название метрики, формула для ее оценки	Исходные данные для вычисления метрики по соответствующей формуле
	реализации (2)	ций, описанных в спецификации требований
Правильность	Точность $X=A/T$	$A$ – количество результатов, подсчитанное пользователями, с уровнем точности, отличающимся от требуемого; $T$ – продолжительность работы
Способность к взаимодействию	Способность к обмену данными (основанная на успешных попытках пользователя) (2)	$A$ – количество случаев, в которых пользователю не удалось обменяться данными с другими ПС или системами; $B$ – количество случаев, в которых пользователь пытался обмениваться данными
Защищенность	Предотвращение разрушения данных (2)	$A$ – количество произошедших случаев разрушения важных данных; $B$ – количество тестовых случаев, направленных на разрушение данных
Соответствие функциональности	Соответствие функциональности (2)	$A$ – количество заданных элементов соответствия функциональности, которые не были выполнены при тестировании; $B$ – общее количество заданных элементов соответствия функциональности
<b>Надежность</b>		
Завершенность	Плотность ошибок $X = A / Z$	$A$ – количество ошибок, обнаруженных в течение определенного испытательного срока; $Z$ – размер продукта
Устойчивость к ошибке	Предотвращение некорректных действий (1)	$A$ – количество предотвращенных критических и серьезных отказов; $B$ – количество выполненных при тестировании тестовых случаев, направленных на проверку типовых некорректных действий, которые могут привести к отказу
Восстанавливаемость	Способность к восстановлению (1)	$A$ – количество случаев успешного восстановления; $B$ – количество случаев восстановления, протестированных согласно требованиям
Соответствие надежности	Соответствие надежности (2)	$A$ – количество заданных элементов соответствия надежности, которые не были выполнены при тестировании; $B$ – общее количество за-



Название подхарактеристики	Название метрики, формула для ее оценки	Исходные данные для вычисления метрики по соответствующей формуле
		данных элементов соответствия надежности
<b>Практичность</b>		
Понятность	Полнота описания (1)	$A$ – количество функций (или классов функций), понятных после прочтения документации на программный продукт; $B$ – общее количество функций (или классов функций), реализуемых программным продуктом
Обучаемость	Эффективность документации пользователя и/или справочной системы (help) (1)	$A$ – количество задач, успешно выполненных после получения оперативной справки и/или чтения документации; $B$ – общее количество протестированных задач
Простота использования	Исправление ошибок при использовании (1)	$A$ – число экранов или форм, где входные данные были успешно модифицированы или изменены (восстановлены) перед очередной обработкой; $B$ – число экранов или форм, где пользователь пытался модифицировать или изменить (восстановить) входные данные в течение испытательного срока использования
Привлекательность	Изменяемость вида интерфейса (1)	$A$ – количество элементов интерфейса, измененных внешне для удовлетворения пользователя; $B$ – количество элементов интерфейса, которые пользователь хочет изменить
Соответствие практичности	Соответствие практичности (2)	$A$ – количество заданных элементов соответствия практичности, которые не были выполнены при тестировании; $B$ – общее количество заданных элементов соответствия практичности
<b>Эффективность</b>		
Поведение во времени	Время отклика $X = A - B$	$A$ – момент времени получения результата; $B$ – момент времени завершения ввода команды
Использование ресурсов	Использование устройств ввода/вывода (1)	$A$ – время занятости устройств ввода/вывода; $B$ – заданное время, предназначенное для использования устройств ввода/вывода
Соответствие эффективно-	Соответствие эффективности	$A$ – количество заданных элементов соответствия эффективности, которые не были выполне-

Название подхарактеристики	Название метрики, формула для ее оценки	Исходные данные для вычисления метрики по соответствующей формуле
сти	(2)	ны при тестировании; <b><i>B</i></b> – общее количество заданных элементов соответствия эффективно-сти
<b><i>Сопровождаемость</i></b>		
Анализируе- мость	Поддержка диагностиче- скими функциями (1)	<b><i>A</i></b> – количество отказов, при которых персонал сопровождения с помощью диагностических функций может диагностировать причину; <b><i>B</i></b> – общее число зарегистрированных отказов
Изменяемость	Возможность управления из- менением ПС (1)	<b><i>A</i></b> – количество фактически записанных данных регистрации изменений; <b><i>B</i></b> – запланированное количество данных регистрации изменений, достаточное для отслеживания изменений ПС
Стабильность	Возникновение отказа после изменения $X = A / N$	<b><i>A</i></b> – количество отказов, возникших в течение заданного периода после устранения отказа; <b><i>N</i></b> – количество отказов, устраненных путем изменения ПС
Тестируемость	Доступность встроенных функций тестирования (1)	<b><i>A</i></b> – количество случаев, в которых персонал сопровождения может использовать встроенные функции тестирования; <b><i>B</i></b> – количество подходящих случаев, в которых можно было бы использовать встроенные тесты
Соответствие сопровождает- мости	Соответствие сопровождает- мости (2)	<b><i>A</i></b> – количество заданных элементов соответствия сопровождаемости, которые не были выполнены при тестировании; <b><i>B</i></b> – общее количество заданных элементов соответствия сопровождаетмости
<b><i>Мобильность</i></b>		
Адаптируе- мость	Адаптируе- мость структур данных (1)	<b><i>A</i></b> – количество работоспособных данных, которые не требуют сопровождения при адаптации; <b><i>B</i></b> – ожидаемое число работоспособных данных в окружающей среде, к которой ПС адаптировано
Настраивае- мость	Простота уста- новки (1)	<b><i>A</i></b> – число успешных случаев приспособливания пользователем операции инсталляции к среде эксплуатации; <b><i>B</i></b> – общее число попыток пользователя приспособить операцию инсталляции

Название подхарактеристики	Название метрики, формула для ее оценки	Исходные данные для вычисления метрики по соответствующей формуле
		к среде окружения
Совместимость	Доступная совместимость $X = A / T$	$A$ – число любых ограничений или непредусмотренных отказов, с которыми пользователь сталкивается во время одновременной работы с другими ПС; $T$ – продолжительность одновременной работы с другими ПС
Взаимозаменяемость	Преемственность данных (1)	$A$ – число данных замещаемого ПС, которые могут продолжать использоваться после его замещения; $B$ – число данных замещаемого ПС, которые по плану должны продолжать использоваться после его замещения
Соответствие мобильности	Соответствие мобильности (2)	$A$ – количество заданных элементов соответствия мобильности, которые не были выполнены при тестировании; $B$ – общее количество заданных элементов соответствия мобильности

Не все метрики, приведенные в данной таблице, удовлетворяют таким критериям обоснованности метрик, как корреляция, трассировка и непротиворечивость. Это касается, например, метрики правильности «Точность», метрики завершенности «Плотность ошибок», метрики поведения во времени «Время отклика», метрики стабильности «Возникновение отказа после изменения», метрики совместимости «Доступная совместимость». Кроме того, данные метрики не удовлетворяют диапазону (3) (см. п. 10.5.1). Это затрудняет их использование при интегральной оценке качества программных продуктов.

#### 10.5.4. Метрики качества программных средств в использовании

**Метрики результативности** оценивают, достигают ли задачи, выполняемые пользователем, заданных целей с точностью и полнотой в заданном контексте использования.

**Метрики продуктивности** оценивают ресурсы, которые затрачивают пользователи в соответствии с достигнутой результативностью в заданном контексте использования.

**Метрики безопасности** оценивают уровень риска причинения вреда людям, бизнесу, программному обеспечению, имуществу или окружающей среде в заданном контексте использования.

**Метрики удовлетворенности** оценивают отношение пользователя к использованию продукта в заданном контексте использования.

Табл. 39 содержит примеры метрик качества в использовании из стандарта *ISO/IEC TR 9126–4:2004* [52]. Во втором столбце таблицы по каждой подхарактеристике приведено название одной метрики, уникальная формула или номер формулы (1) или (2) из подразд. 10.5.1 для оценки данной метрики. Исходные данные в третьем столбце – это данные, используемые в соответствующей формуле для вычисления значения представленной метрики.

Таблица 39

Метрики качества ПС в использовании

Название характеристики	Название метрики, формула для ее оценки	Исходные данные для вычисления метрики по соответствующей формуле
Результативность	Завершение задачи (1)	$A$ – количество завершенных задач; $B$ – общее количество попыток выполнения задач
Продуктивность	Коэффициент продуктивности (1)	$A$ – продуктивное время; $A = B - (B1 + B2 + B3)$ , где $B$ – продолжительность выполнения задачи; $B1$ – продолжительность помощи; $B2$ – продолжительность обработки ошибок; $B3$ – продолжительность поиска
Безопасность	Экономический ущерб (2)	$A$ – число случаев экономического ущерба; $B$ – общее число случаев использования
Удовлетворенность	Использование по собственному усмотрению (1)	$A$ – количество случаев использования заданных в спецификации функций программного средства / приложений / систем; $B$ – количество случаев их запланированного использования

## 10.6. Метод оценки качества программных средств по ISO/IEC 14598–1:1999

Стандарт *ISO/IEC 14598–1:1999* [21] регламентирует метод оценки качества программных средств, который основан на иерархической модели качества, определенной в *ISO/IEC 9126–1:2001* [49].

На рис. 55 приведена схема процесса оценки качества ПС [21]. Процесс оценки состоит из *четырёх стадий*: установка требований к оценке, определение оценки, проектирование оценки и выполнение оценки. Данный процесс может применяться после любой подходящей работы жизненного цикла для

промежуточных или конечного продуктов разработки (сопровождения).

### **Стадия 1. Установка требований к оценке**

Данная стадия состоит из трех этапов.

#### **Этап 1.1. Установка цели оценки**

*Общей целью* оценки качества ПС является поддержка разработки и приобретения ПС, удовлетворяющего заявленные и подразумеваемые потребности пользователей. *Конечная цель* состоит в том, чтобы гарантировать, что продукт обеспечивает требуемое качество.

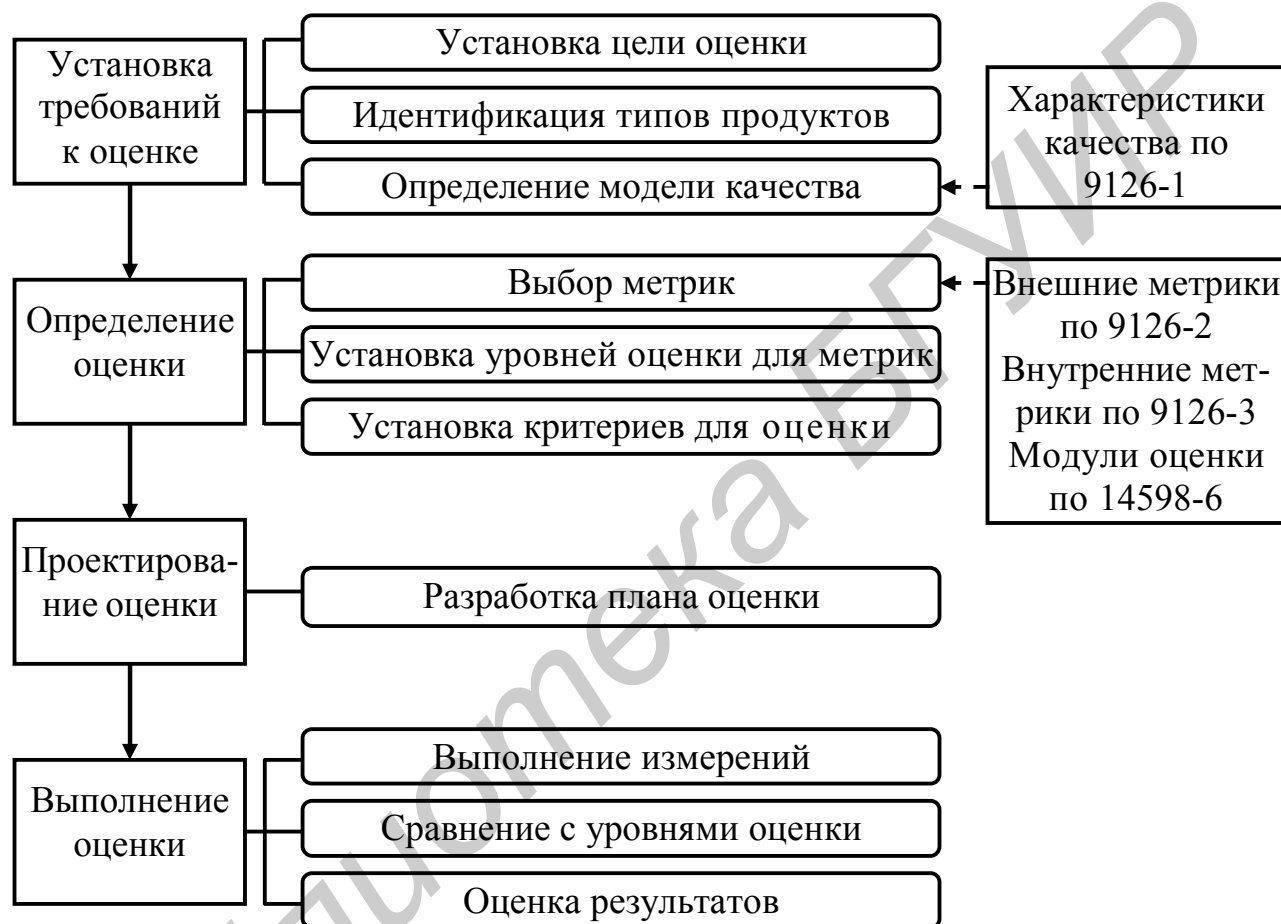


Рис. 55. Процесс оценки качества по ISO/IEC 14598–1:1999

*Целью оценки качества промежуточного продукта* может быть:

- решение о принятии промежуточного продукта от субподрядчика;
- решение о завершении процесса и передаче продукта следующему процессу;

- прогноз или предварительная оценка качества конечного продукта;
- сбор информации о промежуточных продуктах для контроля и управления процессом.

*Целью оценки качества конечного продукта* может быть:

- решение о принятии продукта;
- решение о выпуске продукта;

- сравнение продукта с конкурентными продуктами;
- выбор продукта из числа альтернативных продуктов;
- оценка положительного и отрицательного результата использования продукта;
- решение о сроках улучшения или замены продукта.

Качество программного средства может оцениваться в процессах жизненного цикла, определенных в *ISO/IEC 12207* (см. разд. 2 пособия).

*В процессе заказа ПП* заказчик должен установить требования к внешнему качеству ПП, определить требования к поставщику и оценить качество разработанного ПП по этим требованиям перед его приемкой. Оценка качества ПС заказчиком детально рассмотрена в стандарте *ISO/IEC 14598–4:1999* [24].

При покупке готового программного продукта оценка может использоваться для сравнения альтернативных ПП и гарантии, что выбранный ПП удовлетворяет требованиям по качеству. Оценка качества программного средства оценщиком детально рассмотрена в стандарте *ISO/IEC 14598–5:1998* [25].

*В процессе поставки* поставщик может использовать результаты оценки ПП, чтобы гарантировать, что ПП удовлетворяет требуемым критериям качества как в сравнении с другими ПП, так и установленным заказчиком.

*В процессе разработки* на основании требований к внешнему качеству ПП, установленных заказчиком, разработчик для каждого компонента ПП должен определить спецификацию требований к внешнему качеству в терминах характеристик и подхарактеристик (см. подразд. 10.3 пособия).

На базе спецификации требований к внешнему качеству разработчик должен определить требования к внутреннему качеству ПП. Эти требования должны использоваться на начальных этапах процесса разработки (когда еще нет исполнимых кодов ПП или его компонентов) для проверки качества промежуточных продуктов с целью прогноза и предварительной оценки качества конечного ПП. На последующих этапах процесса разработки (когда уже имеются исполнимые коды модулей, компонентов или всего ПП) должны использоваться требования к внешнему качеству. На их основе должна выполняться оценка внешнего качества промежуточных продуктов и конечного ПП (с целью прогноза итогового внешнего качества в среде эксплуатации). С этой целью обычно используется моделируемая среда с моделируемыми данными. Оценка качества ПС разработчиком рассмотрена в стандарте *ISO/IEC 14598–3:2000* [23].

*В процессе эксплуатации* оценка качества программного средства может использоваться для подтверждения того, что требования к качеству удовлетворяются при различных условиях эксплуатации.

*В процессе сопровождения* оценка программного средства может использоваться для подтверждения того, что требования к качеству по-прежнему удовлетворяются и требования по сопровождаемости и мобильности достигаются.

### *Этап 1.2. Идентификация типов продуктов*

Тип оцениваемого промежуточного или конечного программного продукта зависит от цели оценки и стадии ЖЦ ПС (см. рис. 52). Например, в процессе

разработки интерес представляет оценка промежуточного продукта. На ранних этапах процесса разработки это может быть спецификация требований, архитектура ПП, технический проект ПП, исходные коды модулей. Для них выполняется оценка внутреннего качества с целью прогноза внешнего качества.

На последующих этапах процесса разработки промежуточными продуктами являются исполнимые коды модулей, компонентов и других промежуточных продуктов сборки, а также конечный программный продукт. Для них выполняется оценка внешнего качества в моделируемой среде с моделируемыми данными с целью прогноза внешнего качества ПП в среде эксплуатации.

В процессе эксплуатации оценивается система, частью которой является программный продукт. В этом случае может выполняться, во-первых, оценка внешнего качества при использовании ПС в окружающей среде с целью подтверждения соответствия требованиям к внешнему качеству и прогноза качества ПП в использовании; и, во-вторых, оценка качества в использовании для подтверждения удовлетворения потребностей пользователя в выполнении заданных задач в заданных аппаратных и операционных средах.

### *Этап 1.3. Определение модели качества*

На данном этапе, исходя из типов оцениваемых продуктов, выбирается соответствующая модель качества (внутреннего, внешнего или качества в использовании). Основу модели составляет общая модель качества из стандарта *ISO/IEC 9126-1:2001* (см. рис. 53, 54 пособия). Выбранная модель должна быть адаптирована с учетом целей оценки и конкретных требований к качеству оцениваемого продукта. В процессе адаптации из модели качества стандарта *ISO/IEC 9126-1:2001* должны быть выбраны соответствующие характеристики и подхарактеристики, которые будут оцениваться.

### **Стадия 2. Определение оценки**

Данная стадия также состоит из трех этапов.

#### *Этап 2.1. Выбор метрик*

На данном этапе, исходя из разработанной модели качества программного средства, выбираются соответствующие метрики качества. Каждое измеримое внутреннее или внешнее свойство продукта, влияющее на значение характеристики или подхарактеристики качества, может быть установлено как метрическое. При выборе метрик следует учитывать простоту и экономность их использования. Примеры метрик качества приведены в стандартах *ISO/IEC TR 9126-2, -3, -4* [50 – 52] (см. пп. 10.5.2 – 10.5.4 пособия).

На выбор метрик оказывает влияние также тип требуемых измерений, который зависит от цели оценки. Если целью оценки является исправление недостатков разработки, то для контроля достаточно выполнить несколько измерений продукта. Данные измерения могут быть дополнены мерами из имеющихся контрольных списков или мнениями экспертов.

Если же целью оценки является сравнение готовых программных продуктов или анализ соответствия продукта требованиям к качеству, то необходимо использовать строгие метрики с достаточной точностью измерений. При этом должны быть учтены ошибки в измерениях, вызванные человеческим фактором

или инструментом измерения.

Кроме того, для осуществления прогноза качества программного средства важно, чтобы выбранные внутренние метрики коррелировали с некоторыми аспектами внешнего качества, а выбранные внешние метрики – с аспектами качества в использовании.

#### *Этап 2.2. Установка уровней оценки (ранжирования) для метрик*

Значение, измеренное с помощью метрики, имеет некоторую величину, которая сама по себе не отражает степень удовлетворения результатом измерения. Поэтому шкала измерений должна быть разделена на диапазоны, соответствующие различным степеням удовлетворения требований (см. рис. 51).

#### *Этап 2.3. Установка критериев для оценки*

Чтобы оценить качество продукта, необходимо некоторым образом объединить результаты оценки его различных характеристик. С этой целью должна быть разработана процедура, включающая отдельные критерии для различных характеристик качества. Каждая из характеристик может быть определена в терминах единственной подхарактеристики или средневзвешенной комбинации подхарактеристик. Процедура обычно включает и другие аспекты, такие как время и стоимость, которые вносят вклад в оценку качества программного продукта в конкретной среде окружения.

### **Стадия 3. Проектирование оценки**

Данная стадия состоит из одного этапа.

#### *Этап 3.1. Разработка плана оценки*

В плане оценки должны быть описаны методы оценки и график действий по оценке. Действия по оценке для разработчика, заказчика и оценщика описаны в стандартах *ISO/IEC 14598–3:2000*, *ISO/IEC 14598–4:1999* и *ISO/IEC 14598–5:1998* [23 – 25]. План оценки должен быть согласован с планом количественной оценки качества, регламентированным в *ISO/IEC 14598–2:2000* [22].

### **Стадия 4. Выполнение оценки**

Данная стадия состоит из трех этапов.

#### *Этап 4.1. Выполнение измерений*

Для измерения выбранные метрики применяются к программному продукту. Результатом являются значения на шкалах метрик.

#### *Этап 4.2. Сравнение с уровнями оценки (ранжирование)*

На данном этапе измеренные значения сравниваются с уровнями ранжирования (например, как показано на рис. 51).

#### *Этап 4.3. Оценка результатов*

Оценка результатов – заключительный этап процесса оценки ПС. На данном этапе с учетом решений, принятых при выполнении этапа 2.3, суммируются оцененные уровни метрик. Результатом является заключение о степени удовлетворения продуктом требований к качеству. Полученное в итоге качество сравнивается с другими аспектами, такими как время и стоимость.

На основании оценки и организационных критериев принимается административное решение относительно принятия или отклонения, выпуска или невыпуска программного продукта.



# ВОПРОСЫ ДЛЯ САМОПРОВЕРКИ К ЧАСТИ II

## Раздел 8

1. Назовите обобщенные показатели программных средств, существенные при их разработке и оценке качества.
2. В чем различия внутреннего, внешнего качества и качества в использовании?
3. Что такое контекст использования?
4. Дайте понятие метрики, назовите и опишите виды метрик.
5. Дайте определение оценки и модели качества.
6. Определите понятие ранжирования и уровня ранжирования.
7. Дайте определение уровня качества функционирования.
8. Дайте определение шкалы, перечислите и опишите типы шкал.

## Раздел 9

9. Назовите стандарты в области оценки качества, действующие в Беларуси.
10. Классифицируйте и опишите методы определения показателей качества ПС.
11. Опишите различия иерархических моделей качества программных средств, определенных в ГОСТ 28195–99 и СТБ ИСО/МЭК 9126–2003.
12. Перечислите факторы и критерии качества ПС по ГОСТ 28195–99.
13. Опишите модель Надежности, регламентированную в ГОСТ 28195–99.
14. Опишите метод оценки качества ПС, рекомендованный в ГОСТ 28195–99.
15. Опишите метод оценки качества программных средств, регламентированный в СТБ ИСО/МЭК 9126–2003.
16. Приведите примеры диапазонов ранжирования измеренных значений.

## Раздел 10

17. Назовите действующие международные стандарты в области оценки качества программных средств.
18. Опишите содержание каждой из частей серии стандартов *ISO/IEC 9126–1–4*.
19. Опишите содержание каждой части серии стандартов *ISO/IEC 14598–1–6*.
20. Опишите и представьте графически связь качества ПС с его ЖЦ.
21. Назовите части модели качества ПС, определенные в *ISO/IEC 9126–1:2001*.
22. Назовите уровни иерархической модели качества по *ISO/IEC 9126–1:2001*.
23. Определите каждую из характеристик модели внутреннего и внешнего качества программных средств, определенной в *ISO/IEC 9126–1:2001*.
24. Опишите модель качества в использовании по *ISO/IEC 9126–1:2001*.
25. Перечислите и опишите желательные свойства метрик качества ПС.
26. Перечислите и опишите критерии обоснованности метрик качества ПС.
27. По каждой характеристике качества приведите примеры внутренних, внешних метрик и метрик качества в использовании.
28. Опишите метод оценки качества программных средств, определенный в *ISO/IEC 14598–1:1999*.

# **ЧАСТЬ III**

## **СЕРТИФИКАЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

Библиотека eLibrary

# 11. СЕРТИФИКАЦИЯ ПРОГРАММНЫХ СРЕДСТВ

## 11.1. Основные понятия и определения в области технического нормирования, стандартизации и оценки соответствия

5 января 2004 г. в Республике Беларусь приняты и введены в действие Законы № 262-З «О техническом нормировании и стандартизации» и № 269-З «Об оценке соответствия требованиям технических нормативных правовых актов в области технического нормирования и стандартизации» [54, 55].

Данные законы разработаны с учетом соглашений Всемирной торговой организации (ВТО). Они базируются на практике Европейского Союза (ЕС), определенной в Директивах ЕС Нового и Глобального подхода [56, 57], и направлены на обеспечение единой государственной политики в области технического нормирования, стандартизации и оценки соответствия.

Первый из законов регулирует отношения, возникающие при разработке, утверждении и применении технических требований к продукции, процессам ее разработки, производства, эксплуатации (использования), хранения, перевозки, реализации и утилизации (именуемым далее процессами ее жизненного цикла) или к оказанию услуг, определяет правовые и организационные основы технического нормирования и стандартизации.

Второй закон определяет правовые и организационные основы оценки соответствия объектов оценки требованиям технических нормативных правовых актов в области технического нормирования и стандартизации и направлен на совершенствование механизма оценки в области подтверждения соответствия и аккредитации с учетом международных принципов и требований Соглашения по техническим барьерам в торговле ВТО.

В Законе «О техническом нормировании и стандартизации» определены следующие **технические нормативные правовые акты** в области технического нормирования и стандартизации:

- технические регламенты;
- технические кодексы установившейся практики;
- государственные стандарты Республики Беларусь;
- технические условия.

**Технический регламент (ТР)** – это технический нормативный правовой акт, разработанный в процессе технического нормирования, устанавливающий

непосредственно и/или путем ссылки на технические кодексы установившейся практики и/или государственные стандарты Республики Беларусь *обязательные для соблюдения технические требования*, связанные с безопасностью продукции и процессов ее жизненного цикла.

**Технический кодекс установившейся практики (ТКП)** – это технический нормативный правовой акт, разработанный в процессе стандартизации, содержащий *основанные на результатах установившейся практики технические требования к процессам жизненного цикла* продукции или оказанию услуг.

**Стандарт** – это технический нормативный правовой акт, разработанный в процессе стандартизации *на основе согласия большинства заинтересованных субъектов технического нормирования и стандартизации* и содержащий технические требования к продукции, процессам ее жизненного цикла или оказанию услуг.

**Технические условия (ТУ)** – это технический нормативный правовой акт, разработанный в процессе стандартизации, утвержденный юридическим лицом или индивидуальным предпринимателем и *содержащий технические требования к конкретным типу, марке, модели, виду реализуемой ими продукции* или оказываемой услуге, включая правила приемки и методы контроля.

В Законе [54] определено, что *обязательные* требования устанавливаются в технических регламентах, принимаемых органами власти, а стандарты применяются в *добровольном* порядке.

В области оценки соответствия приняты следующие **основные термины и их определения** [55].

**Аккредитация:** вид оценки соответствия, результатом осуществления которого является официальное признание компетентности юридического лица в выполнении работ по подтверждению соответствия и (или) проведении испытаний продукции.

**Аккредитованная испытательная лаборатория (центр):** юридическое лицо, аккредитованное для проведения испытаний продукции в определенной области аккредитации.

**Аккредитованный орган по сертификации:** юридическое лицо, аккредитованное для выполнения работ по подтверждению соответствия в определенной области аккредитации.

**Аттестат аккредитации:** документ, удостоверяющий компетентность юридического лица в выполнении работ по подтверждению соответствия или в проведении испытаний продукции в определенной области аккредитации.

**Декларация о соответствии:** документ, в котором изготовитель (продавец) удостоверяет соответствие производимой и (или) реализуемой им продукции требованиям технических нормативных правовых актов в области технического нормирования и стандартизации.

**Декларирование соответствия:** форма подтверждения соответствия, осуществляемого изготовителем (продавцом).

**Национальная система подтверждения соответствия Республики Беларусь:** установленная совокупность субъектов оценки соответствия, норматив-

ных правовых актов и технических нормативных правовых актов в области технического нормирования и стандартизации, определяющих правила и процедуры подтверждения соответствия и функционирования системы в целом.

*Область аккредитации:* сфера деятельности, в которой аккредитованному органу по сертификации или аккредитованной испытательной лаборатории (центру) предоставлено право на выполнение работ по подтверждению соответствия или проведение испытаний продукции.

*Оценка соответствия:* деятельность по определению соответствия объектов оценки соответствия требованиям технических нормативных правовых актов в области технического нормирования и стандартизации.

*Подтверждение соответствия:* вид оценки соответствия, результатом осуществления которого является документальное удостоверение соответствия объекта оценки требованиям технических нормативных правовых актов в области технического нормирования и стандартизации.

*Сертификат компетентности:* документ, удостоверяющий профессиональную компетентность физического лица в выполнении определенных работ, услуг.

*Сертификат соответствия:* документ, удостоверяющий соответствие объекта оценки требованиям технических нормативных правовых актов в области технического нормирования и стандартизации.

*Сертификация:* форма подтверждения соответствия, осуществляемого аккредитованным органом по сертификации.

*Система управления качеством (система менеджмента качества):* часть общей системы управления, включающая организационную структуру, планирование, ответственность, методы, процедуры, процессы, ресурсы, необходимые для обеспечения качества продукции и (или) услуг; как правило, система управления качеством является частью системы управления предприятием или организацией.

*Схема подтверждения соответствия:* установленная последовательность действий, результаты которых рассматриваются в качестве доказательств соответствия объекта оценки требованиям технических нормативных правовых актов в области технического нормирования и стандартизации.

*Форма подтверждения соответствия:* установленный порядок документального удостоверения соответствия объекта оценки требованиям технических нормативных правовых актов в области технического нормирования и стандартизации.

## **11.2. Общие сведения об оценке соответствия в Республике Беларусь**

Оценка соответствия в Беларуси выполняется на основе *Закона Республики Беларусь № 269-З «Об оценке соответствия требованиям технических нор-*

мативных правовых актов в области технического нормирования и стандартизации» [55]. Оценка соответствия осуществляется в целях:

- обеспечения защиты жизни, здоровья и наследственности человека, имущества и охраны окружающей среды;
- повышения конкурентоспособности продукции (услуг);
- создания благоприятных условий для обеспечения свободного перемещения продукции на внутреннем и внешнем рынках.

Принципами оценки соответствия являются:

- гармонизация с международными и межгосударственными (региональными) подходами в области оценки соответствия;
- обеспечение идентичности правил и процедур подтверждения соответствия продукции отечественного и иностранного производства;
- соблюдение требований конфиденциальности сведений, полученных при выполнении работ по оценке соответствия.

Основными объектами оценки соответствия являются:

- продукция;
- процессы жизненного цикла продукции (разработка, производство, эксплуатация, хранение, перевозка, реализация и утилизация);
- оказание услуг;
- система управления качеством;
- система управления окружающей средой;
- юридические лица;
- персонал.

Субъектами оценки соответствия являются:

- государство в лице уполномоченных государственных органов;
- аккредитованные органы по сертификации;
- аккредитованные испытательные лаборатории (центры);
- заявители на аккредитацию;
- заявители на подтверждение соответствия.

Законом [55] предусмотрено два вида оценки соответствия: аккредитация и подтверждение соответствия (рис. 56).

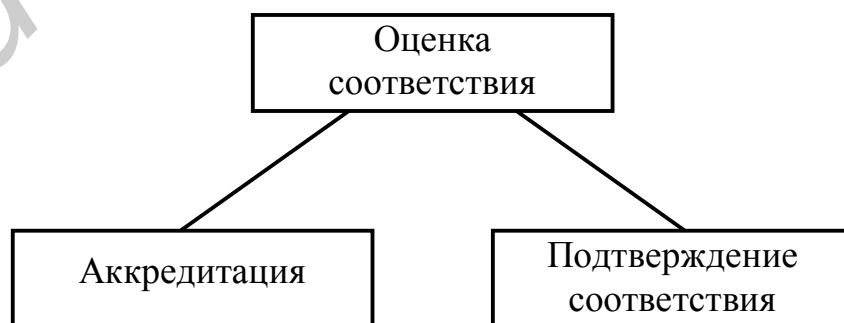


Рис. 56. Виды оценки соответствия

К документам об оценке соответствия относятся:

- аттестат аккредитации;
- сертификат соответствия;
- декларация о соответствии;
- сертификат компетентности.

Систему подтверждения соответствия в Республике Беларусь иллюстрирует рис. 57 [59].



Рис. 57. Система подтверждения соответствия в Республике Беларусь

Национальным органом по оценке соответствия Республики Беларусь является Государственный комитет по стандартизации, метрологии и сертификации при Совете министров Республики Беларусь (Госстандарт). К органам по

сертификации относится, например, Белорусский государственный институт стандартизации и сертификации (БелГИСС).

Подтверждение соответствия осуществляется в *целях*:

- удостоверения соответствия объектов оценки требованиям технических нормативных правовых актов в области технического нормирования и стандартизации;

- содействия потребителям в компетентном выборе продукции (услуг).

*Принципами подтверждения соответствия* являются:

- открытость и доступность правил и процедур подтверждения соответствия;

- независимость аккредитованных органов по сертификации и аккредитованных испытательных лабораторий (центров) от заявителей на подтверждение соответствия;

- минимизация сроков исполнения и затрат на прохождение процедур подтверждения соответствия;

- защита имущественных интересов заявителей на подтверждение соответствия, в том числе путем соблюдения конфиденциальности в отношении сведений, полученных в процессе прохождения процедур подтверждения соответствия;

- недопустимость ограничения конкуренции при выполнении работ по подтверждению соответствия и проведении испытаний продукции.

Порядок выполнения работ по подтверждению соответствия устанавливается в Национальной системе подтверждения соответствия Республики Беларусь. Положительные результаты подтверждения соответствия удостоверяются одним из следующих *документов*:

- *сертификатом соответствия* или *сертификатом компетентности*, выдаваемым аккредитованным органом по сертификации заявителю на подтверждение соответствия;

- *декларацией о соответствии*, принятой заявителем на подтверждение соответствия и зарегистрированной аккредитованным органом по сертификации.

Подтверждение соответствия может носить **обязательный** или **добровольный характер**.

*Обязательное подтверждение соответствия* осуществляется в формах:

- обязательной сертификации;
- декларирования соответствия.

*Добровольное подтверждение соответствия* осуществляется только в форме добровольной сертификации.

Обязательному подтверждению соответствия подлежат объекты оценки, включенные в **Перечень** продукции, услуг, персонала и других объектов оценки соответствия, подлежащих обязательному подтверждению соответствия в Республике Беларусь. В данном Перечне устанавливаются:

- виды продукции, услуг, персонал и иные объекты оценки соответствия,



подлежащие обязательному подтверждению соответствия;

- технические нормативные правовые акты в области технического нормирования и стандартизации, на соответствие которым проводится обязательное подтверждение соответствия;

- формы обязательного подтверждения соответствия.

*Критериями при формировании Перечня являются:*

- потенциальная опасность продукции, услуг, деятельности персонала и функционирования других объектов оценки соответствия для жизни, здоровья и наследственности человека, имущества и окружающей среды;

- несовместимость технических средств отечественного и иностранного производства.

**Обязательная сертификация** осуществляется аккредитованным органом по сертификации на основе договора с заявителем. Схемы подтверждения соответствия при обязательной сертификации зависят от видов объектов оценки. Данные схемы должны устанавливаться соответствующим техническим регламентом. Если схемой установлена необходимость проведения испытаний продукции, то они проводятся аккредитованной испытательной лабораторией (центром) на основе договора с заявителем.

**Декларирование соответствия** осуществляется заявителем на подтверждение соответствия *только в отношении продукции* одним из следующих способов:

- путем принятия декларации о соответствии на основании собственных доказательств;

- путем принятия декларации о соответствии на основании собственных доказательств и доказательств, полученных с участием аккредитованного органа по сертификации или аккредитованной испытательной лаборатории.

Схемы подтверждения соответствия при декларировании соответствия различных видов продукции должны устанавливаться соответствующим техническим регламентом. Если схемой установлена необходимость проведения испытаний продукции, то они проводятся аккредитованной испытательной лабораторией (центром) на основе договора с заявителем.

**Добровольная сертификация** осуществляется аккредитованным органом по сертификации по инициативе заявителя на подтверждение соответствия на основе договора. При добровольной сертификации заявитель самостоятельно выбирает технические нормативные правовые акты, на соответствие которым осуществляется сертификация, и определяет номенклатуру контролируемых показателей. В номенклатуру показателей обязательно должны включаться показатели безопасности, если они установлены в соответствующих технических нормативных правовых актах.

*Форма, правила и процедуры* обязательного подтверждения соответствия, а также правила и процедуры добровольной сертификации устанавливаются в документах Национальной системы подтверждения соответствия Республики Беларусь (НСПС РБ).

К основополагающим документам НСПС РБ относятся:

- *ТКП 5.1.01–2004 НСПСРБ. Основные положения.* Данный ТКП является основным в комплексе ТКП и стандартов, обеспечивающих функционирование НСПС РБ. ТКП определяет общие правила организации работ по подтверждению соответствия, структуру НСПС РБ и функции ее органов.
- *ТКП 5.1.02–2004 НСПС РБ. Порядок сертификации продукции. Основные положения.* ТКП устанавливает общие требования к порядку проведения сертификации отечественной и импортируемой продукции и применяется при обязательной и добровольной сертификации продукции. На его основе разрабатываются и применяются совместно с ним порядки сертификации групп однородной продукции, учитывающие особенности ее ЖЦ.
- *ТКП 5.1.03–2004 НСПС РБ. Порядок декларирования соответствия продукции. Основные положения.* ТКП устанавливает порядок проведения декларирования соответствия отечественной и импортируемой продукции требованиям технических нормативных правовых актов в области технического нормирования и стандартизации.
- *ТКП 5.1.04–2004 НСПС РБ. Порядок сертификации услуг. Основные положения.* ТКП устанавливает общие требования к порядку сертификации услуг в рамках НСПС РБ. Данный ТКП применяется при проведении сертификации услуг и разработке порядка сертификации конкретного вида услуг.
- *ТКП 5.1.05–2004 НСПС РБ. Порядок сертификации систем менеджмента качества. Основные положения.* ТКП устанавливает общие требования к порядку проведения работ по сертификации систем менеджмента качества на соответствие требованиям технических нормативных правовых актов (СТБ ИСО 9001 и др.). Требования ТКП могут использоваться органами по сертификации для разработки порядка проведения сертификации систем менеджмента качества организаций.
- *ТКП 5.1.06–2004 НСПС РБ. Порядок сертификации компетентности персонала. Основные положения.* ТКП устанавливает общие требования к порядку проведения сертификации профессиональной компетентности персонала, осуществляющего конкретные виды работ (услуг) в определенной области деятельности. Требования ТКП применяются органами по сертификации при разработке порядка проведения сертификации персонала.
- *ТКП 5.1.09–2004 НСПС РБ. Порядок сертификации экспертов–аудиторов по качеству.* ТКП устанавливает требования, предъявляемые к экспертам–аудиторам по качеству в НСПС РБ, права, обязанности, ответственность и порядок их сертификации.
- *ТКП 5.1.11–2004 НСПС РБ. Порядок применения форм и схем подтверждения соответствия требованиям технических нормативных правовых актов в области технического нормирования и стандартизации.* ТКП устанавливает порядок выбора и применения форм и схем обязательного подтверждения соответствия продукции требованиям технических нормативных правовых актов при разработке технических регламентов.

## 11.3. Организация сертификации программных средств

К программным средствам может применяться такая форма подтверждения соответствия, как сертификация.

Отличием процесса сертификации программных средств от сертификации других видов продукции является высокая сложность. Связано это, в первую очередь, с невозможностью провести исчерпывающее тестирование сертифицируемых ПС, имеющих, как правило, достаточно большой объем.

Существует *два вида сертификации программных средств* – обязательная сертификация и добровольная сертификация. Результатом положительного прохождения сертификации является выдача *сертификата соответствия*.

**Обязательной сертификации** подвергаются программные средства, выполняющие особо ответственные функции, в которых недостаточное качество и ошибки представляют потенциальную опасность для жизни, здоровья и наследственности человека, имущества и окружающей среды.

**Добровольная сертификация** применяется для удостоверения качества программного средства с целью повышения их конкурентоспособности и создания благоприятных условий для обеспечения свободного перемещения продукции на внутреннем и внешнем рынках.

*Методология принятия решений о допустимости выдачи сертификата* на программное средство основывается на оценке степени его соответствия действующим и/или специально разработанным документам:

- соответствующим международным, межгосударственным и национальным стандартам;
- стандартам на сопровождающую документацию;
- нормативным и эксплуатационным документам на конкретное программное средство: техническим условиям, техническим описаниям, спецификациям требований и другим регламентирующим документам.

**Процесс сертификации** программных средств включает [58]:

- 1) анализ и выбор разработчиком или заказчиком компетентного в данной области органа по сертификации;
- 2) подачу заявителем заявки на сертификацию в орган сертификации;
- 3) принятие органом сертификации решения по заявке, выбор испытательной лаборатории и схемы сертификации;
- 4) обзор и идентификацию версий ПС, подлежащих испытаниям;
- 5) сертификационные испытания ПС испытательной лабораторией;
- 6) анализ полученных результатов и принятие решения лабораторией или органом сертификации о возможности выдачи заявителю сертификата соответствия;
- 7) выдачу органом сертификации заявителю сертификата на сертифицированную версию программного средства;

8) осуществление инспекционного контроля органом сертификации за сертифицированным программным средством;

9) проведение заявителем корректирующих мероприятий при нарушении соответствия программного средства установленным требованиям;

10) регистрацию и публикацию информации о результатах сертификации программного средства.

Международными стандартами определены состав и содержание документов, поддерживающих организацию сертификации программных средств.

В *состав документов заявителя* входят:

- заявка на проведение сертификации;
- проект договора на сертификационные испытания;
- программное средство;
- комплект технической документации, включая техническое задание или спецификацию требований и эксплуатационную документацию на программное средство и его компоненты.

В *состав документов органа сертификации* входят:

- регистрационная карта сертифицируемого объекта;
- заключение по результатам рассмотрения заявки на сертификацию;
- задание на проведение сертификации и требования к ней;
- план сертификационных испытаний;
- заключение по результатам сертификационных испытаний;
- оформленный сертификат соответствия.

В *состав документов испытательной лаборатории* входят:

- характеристики объекта испытаний;
- комплект технической документации;
- действующие международные, государственные и ведомственные стандарты на разработку и испытания программных средств и на техническую документацию;
- программа сертификационных испытаний по всем требованиям технического задания и положениям документации;
- методика сертификационных испытаний по каждому разделу требований технического задания и документации;
- инструментальные средства и методы испытаний;
- регистрационная карта сертификационных испытаний;
- протоколы сертификационных испытаний;
- отчет о проведенных испытаниях и предложение о выдаче сертификата.

Базовыми стандартами, используемыми испытательной лабораторией в процессе оценки качества программного средства и его соответствия требованиям к качеству, являются стандарты *ISO/IEC 14598–1:1999* и *ISO/IEC 14598–5:1998* [21, 25]. Общий процесс оценки, регламентированный в данных стандартах, подробно рассмотрен в подразд. 10.6 пособия.

# ВОПРОСЫ ДЛЯ САМОПРОВЕРКИ К ЧАСТИ III

## Раздел 11

1. Назовите законы в области технического нормирования, стандартизации и оценки соответствия, действующие в Республике Беларусь.
2. Перечислите и дайте определения технических нормативных правовых актов в области технического нормирования и стандартизации, существующих на территории Республики Беларусь в соответствии с действующим законодательством.
3. Что такое оценка соответствия?
4. Назовите виды оценки соответствия.
5. Назовите цели оценки соответствия.
6. Перечислите принципы оценки соответствия.
7. Что может являться объектами оценки соответствия?
8. Перечислите субъекты оценки соответствия.
9. Перечислите виды документов об оценке соответствия.
10. Перечислите принципы подтверждения соответствия.
11. Какой характер может носить подтверждение соответствия в Республике Беларусь?
12. Назовите формы подтверждения соответствия в Республике Беларусь и охарактеризуйте каждую из них.
13. Что является критерием при формировании Перечня продукции, услуг, персонала и других объектов оценки соответствия, подлежащих обязательному подтверждению соответствия в Республике Беларусь?
14. Какие документы относятся к основополагающим документам Национальной системы подтверждения соответствия Республики Беларусь? Приведите примеры.
15. Назовите форму подтверждения соответствия, которая может применяться к программным средствам. Перечислите ее виды.
16. Перечислите документы, на степень соответствия которым может оцениваться программное средство при его сертификации.
17. Опишите процесс сертификации программных средств.
18. Перечислите документы заявителя, необходимые для организации сертификации программных средств.
19. Перечислите документы органа сертификации, поддерживающие организацию сертификации программных средств.
20. Перечислите документы испытательной лаборатории, поддерживающие организацию сертификации программных средств.
21. Назовите базовые стандарты, используемые испытательной лабораторией в качестве основы процесса оценки качества программного средства и его соответствия требованиям к качеству.

## ЛИТЕРАТУРА

1. ГОСТ 28195–89. Оценка качества программных средств. Общие положения.
2. ГОСТ 28195–99. Оценка качества программных средств. Общие положения.
3. ГОСТ 28806–90. Качество программных средств. Термины и определения.
4. ГОСТ Р ИСО/МЭК 12119–2000. Информационная технология. Пакеты программ. Требования к качеству и тестирование.
5. ГОСТ Р ИСО/МЭК 12207–99. Информационная технология. Процессы жизненного цикла программных средств.
6. ГОСТ Р ИСО/МЭК 14764–2002. Информационная технология. Сопровождение программных средств.
7. ГОСТ Р ИСО/МЭК 15910–2002. Информационная технология. Процесс создания документации пользователя программного средства.
8. ГОСТ Р ИСО/МЭК ТО 15271–2002. Информационная технология. Руководство по применению ГОСТ Р ИСО/МЭК 12207 (Процессы жизненного цикла программных средств).
9. ГОСТ Р ИСО/МЭК ТО 16326–2002. Программная инженерия. Руководство по применению ГОСТ Р ИСО/МЭК 12207 при управлении проектом.
10. ГОСТ Р ИСО 9001–96. Системы качества. Модель обеспечения качества при проектировании, разработке, производстве, монтаже и обслуживании.
11. ГОСТ Р ИСО/МЭК 9126–93. Информационная технология. Оценка программной продукции. Характеристики качества и руководства по их применению.
12. СТБ ИСО/МЭК 12207–2003. Информационная технология. Процессы жизненного цикла программных средств.
13. СТБ ИСО/МЭК 14764–2003. Информационная технология. Сопровождение программных средств.
14. СТБ ИСО/МЭК 9126–2003. Информационные технологии. Оценка программной продукции. Характеристики качества и руководства по их применению.
15. IEEE 1517–1999. Стандарт IEEE по информационной технологии – Процессы жизненного цикла программных средств – Процессы повторного использования.
16. ISO 10006–1997. Управление качеством – Руководства по качеству при управлении проектом.
17. ISO/IEC 12207:1995. Информационная технология – Процессы жизненного цикла программных средств.
18. ISO/IEC 12207:1995. Информационная технология – Процессы жизненного цикла программных средств / Amd.1:2002.
19. ISO/IEC 12207:1995. Информационная технология – Процессы жизнен-

ного цикла программных средств / Amd.2:2004.

20. ISO 13407:1999. Эргономика – Эргономика взаимодействия человека с системой – Процесс проектирования диалоговых систем, ориентированный на человека.

21. ISO/IEC 14598–1:1999. Информационная технология – Оценка программного продукта – Часть 1: Общий обзор.

22. ISO/IEC 14598–2:2000. Программная инженерия – Оценка продукта – Часть 2: Планирование и управление.

23. ISO/IEC 14598–3:2000. Программная инженерия – Оценка продукта – Часть 3: Процесс для разработчиков.

24. ISO/IEC 14598–4:1999. Программная инженерия – Оценка продукта – Часть 4: Процесс для заказчиков.

25. ISO/IEC 14598–5:1998. Информационная технология – Оценка программного продукта – Часть 5: Процесс для оценщиков.

26. ISO/IEC 14598–6:2001. Программная инженерия – Оценка продукта – Часть 6: Документация модулей оценки.

27. ISO/IEC 14764:1999. Информационная технология – Сопровождение программных средств.

28. ISO/IEC TR 15271:1998. Информационная технология – Руководство по применению ISO/IEC 12207 (Процессы жизненного цикла программных средств).

29. ISO/IEC 15288:2002. Системная инженерия – Процессы жизненного цикла системы.

30. ISO/IEC 15504–1:2004. Информационная технология – Оценка процессов программных средств – Часть 1: Концепции и словарь.

31. ISO/IEC 15504–2:2003. Информационная технология – Оценка процессов – Часть 2: Проведение оценки.

32. ISO/IEC 15504–3:2004. Информационная технология – Оценка процессов – Часть 3: Руководство по проведению оценки.

33. ISO/IEC TR 15504–4:1998. Информационная технология – Оценка процессов программных средств – Часть 4: Руководство по проведению оценки.

34. ISO/IEC TR 15504–5:1999. Информационная технология – Оценка процессов программных средств – Часть 5: Модель оценки и руководящие указания.

35. ISO/IEC TR 15504–6:1998. Информационная технология – Оценка процессов программных средств – Часть 6: Квалификация и обучение экспертов-оценщиков.

36. ISO/IEC TR 15504–7:1998. Информационная технология – Оценка процессов программных средств – Часть 7: Руководство по использованию при усовершенствовании процессов.

37. ISO/IEC TR 15504–8:1998. Информационная технология – Оценка процессов программных средств – Часть 8: Руководство по использованию при определении зрелости процессов поставщика.

38. ISO/IEC TR 15504–9:1998. Информационная технология – Оценка про-

цессов программных средств – Часть 9: Словарь.

39. ISO/IEC 15846:1998. Информационная технология – Процессы жизненного цикла программных средств – Управление конфигурацией.

40. ISO/IEC 15910:1999. Информационная технология – Процесс создания документации пользователя программного средства.

41. ISO/IEC 15939:2002. Программная инженерия – Процесс измерения программных средств.

42. ISO/IEC 16085:2004. Информационная технология – Процессы жизненного цикла программных средств – Управление рисками.

43. ISO/IEC TR 16326:1999. Программная инженерия – Руководство по применению ISO/IEC 12207 при управлении проектом.

44. ISO/IEC TR 19760:2003. Системная инженерия – Руководство по использованию ISO/IEC 15288 (Процессы жизненного цикла системы).

45. ISO/IEC 2382-1:1993. Информационные технологии – Словарь – Часть 1: Основополагающие термины.

46. ISO/IEC 2382-20:1990. Информационные технологии – Словарь – Часть 20: Разработка систем.

47. ISO 8402:1994. Управление качеством и обеспечение качества. Словарь.

48. ISO/IEC 9126:1991. Информационная технология – Оценка программного продукта – Характеристики качества и руководства по их применению.

49. ISO/IEC 9126-1:2001. Программная инженерия – Качество продукта – Часть 1: Модель качества.

50. ISO/IEC TR 9126-2:2003. Программная инженерия – Качество продукта – Часть 2: Внешние метрики.

51. ISO/IEC TR 9126-3:2003. Программная инженерия – Качество продукта – Часть 3: Внутренние метрики.

52. ISO/IEC TR 9126-4:2004. Программная инженерия – Качество продукта – Часть 4: Метрики качества в использовании.

53. Руководство PMBOK. Руководство по сведениям для органа управления проектом. – Комитет по стандартам института управления проектом PMI (США). – 1996.

54. Закон Республики Беларусь от 5 января 2004 г. № 262-З «О техническом нормировании и стандартизации».

55. Закон Республики Беларусь от 5 января 2004 г. № 269-З «Об оценке соответствия требованиям технических нормативных правовых актов в области технического нормирования и стандартизации».

56. Council Resolution of 7 May 1985 on a new approach to technical harmonization and standards.

57. Council Resolution of 21 December 1989 on a global approach to conformity assessment.

58. Липаев В.В. Обеспечение качества программных средств. Методы и стандарты. – М.: СИНТЕГ, 2001.

59. [www.gosstandart.gov.by](http://www.gosstandart.gov.by)



Учебное издание

**Бахтизин Вячеслав Вениаминович,  
Глухова Лилия Александровна**

**Стандартизация и сертификация  
программного обеспечения**

Учебное пособие

Редактор *Т.Н.Крюкова*  
Корректор *Е.Н.Батурчик*  
Компьютерная верстка *Е.Н. Мирошниченко*

Подписано в печать 13.02.2006. Формат 60x84 1/16. Бумага офсетная. Гарнитура «Таймс».  
Печать ризографическая. Усл. печ. л. 11,63. Уч.-изд. л. 8,5. Тираж 200 экз. Заказ 115.

Издатель и полиграфическое исполнение: Учреждение образования  
«Белорусский государственный университет информатики и радиоэлектроники»  
Лицензия на осуществление издательской деятельности №02330/0056964 от 01.04.2004.  
Лицензия на осуществление полиграфической деятельности №02330/0131518 от 30.04.2004.  
220013, Минск, П. Бровки, 6