ГОСУДАРСТВЕННЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Информационная технология

КЛАССИФИКАЦИЯ ПРОГРАММНЫХ СРЕДСТВ

Издание официальное

БЗ 10—2000/303

ГОССТАНДАРТ РОССИИ
Москва

gost.e-ecolog.ru

Предисловие

1 РАЗРАБОТАН И ВНЕСЕН Всероссийским научно-исследовательским институтом стандартизации (ВНИИстандарт) Госстандарта России

2 ПРИНЯТ И ВВЕДЕН В ДЕЙСТВИЕ Постановлением Госстандарта России от 11 июня 2002 г. № 238-ст

3 Настоящий стандарт содержит полный аутентичный текст международного стандарта ИСО/МЭК ТО 12182—98«Информационная технология. Классификация программных средств»

4 ВВЕДЕН ВПЕРВЫЕ

© ИПК Издательство стандартов, 2002

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Госстандарта России

Содержание

Введение . . . IV
1 Область применения . . . 1
1.1 Применение . . . 1
1.2 Круг пользователей и назначение . . . 1
1.3 Ограничения . . . 1
2 Соответствие . . . 1
3 Нормативные ссылки . . . 2
4 Определения . . . 2
5 Сокращения . . . 2
6 Концепция классификации ПС . . . 2
6.1 Структура видов . . . 3
6.2 Выбор видов и классов . . . 3
7 Схема классификации . . . 3
7.1 Функция ПС . . . 4
7.2 Прикладная область информационной системы . . . 4
7.3 Режим эксплуатации . . . 4
7.4 Масштаб ПС . . . 4
7.5 Представление данных . . . 4
7.6 Исходный язык . . . 5
7.7 Критичность ПС . . . 5
7.8 Класс пользователя . . . 5
7.9 Стабильность ПС . . . 5
7.10 Готовность программного продукта . . . 5
7.11 Использование программных данных . . . 5
7.12 Требуемые рабочие характеристики . . . 6
7.13 Требование защиты . . . 6
7.14 Требование надежности . . . 6
7.15 Вычислительная система и среда . . . 6
7.16 Требование к вычислительным ресурсам . . . 7
8 Применение схемы . . . 7
8.1 Применение к области действия стандартов . . . 7
8.2 Применение к стандартам . . . 7
8.3 Применение к программным пакетам . . . 8
8.4 Пример классификации. Пакет для обработки текста . . . 8

Введение

Настоящий стандарт предназначен для специалистов в области программной инженерии, пользователей и разработчиков стандартов в данной области.

Специалистам в области программной инженерии настоящий документ должен помочь в определении вида (типа) программного средства, для которого применимы конкретные стандарты программной инженерии, установлении критериев запланированного риска, определения соответствия применяемой модели жизненного цикла (ЖЦ) условиям реализации конкретного проекта, определении усилия, необходимых для конкретной фазы жизненного цикла, и соответствующего для нее инструментария.

Пользователям и разработчикам стандартов настоящий документ должен помочь в определении подходов к классификации программных средств (ПС) и возможных вариантов стандартов для них на базе принятой схемы классификации и в последующем использовании данной схемы применительно к соответствующим ПС и стандартам программной инженерии.

С целью пояснения описываемой схемы классификации ПС в настоящем документе приведены примеры. Разработчикам стандартов настоящий документ поможет при классификации существующих, разрабатываемых и перспективных стандартов.

Настоящий стандарт определяет основы классификации ПС, схему классификации и содержит примеры применения соответствующих стандартов.

к ГОСТ Р ИСО/МЭК ТО 12182—2002 Информационная технология. Классификация программных средств

В каком месте	Напечатано	Должно быть
С. 1. Дата введения	2007—07—01	2003—07—01

(ИУС № 12 2002 г.)

ГОСУДАРСТВЕННЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Информационная технология

КЛАССИФИКАЦИЯ ПРОГРАММНЫХ СРЕДСТВ

Information technology. Categorization of software

Дата введения 2007—07—01

1 Область применения

Областью применения настоящего стандарта является классификация программных средств (включая соответствующие продукты разработки программных средств и данные), создаваемых в процессах программной инженерии. Стандарт описывает схему классификации программных средств (ПС), охватывающую различные подходы к их классификации, а также существенные характеристики и атрибуты, описывающие и определяющие данные ПС и их классы.

1.1 Применение

Классификация ПС, установленная в настоящем стандарте, распространяется на область программной инженерии и связанные с ней стандарты, ПС, данные и методологии.

1.2 Круг пользователей и назначение

Настоящий стандарт в основном предназначен для специалистов в области программной инженерии, пользователей и разработчиков стандартов в данной области.

Установленная в стандарте классификация ПС предназначена или применима для определения классов ПС, стандартов программной инженерии, используемых для конкретных ПС, и связей программных задач, процессов или продуктов со стандартами программной инженерии.

Использование установленной классификации ПС заключается в создании соответствующего классификационного элемента для каждого вида, установленного в разделе 7 настоящего стандарта, применительно к рассматриваемому ПС или подходящему стандарту программной инженерии. В некоторых случаях допускается наличие пустого элемента.

Часто возникают ситуации, когда процессы программной инженерии и продукты данных процессов применяются для приобретения или разработки некоторых разновидностей ПС. Например ИСО/МЭК 6592 применяют для больших прикладных систем, а ГОСТ Р ИСО/МЭК 9127 – для пакетов программ. В настоящем стандарте установлена схема классификации, помогающая:

1) уточнить области применения используемого стандарта или ПС;
2) определить и выбрать стандарты, применимые к прикладному ПС;
3) определить классификационные характеристики новых стандартов.

1.3 Ограничения

Так как программная инженерия является быстро изменяющейся областью деятельности, описанная в настоящем стандарте классификация может служить только в качестве концептуальной схемы. Поэтому пользователи должны применять собственные подходы к использованию данной классификации. Приведенная в настоящем стандарте схема классификации является эмпирической. Ее описание не основано на четко установленных потребностях пользователей. Поэтому применение данной схемы в практической деятельности не является обязательным.

2 Соответствие

Не установлено.

Издание официальное

3 Нормативные ссылки

В настоящем стандарте использованы ссылки на следующие стандарты:

ГОСТ Р ИСО/МЭК 9126—93 Информационная технология. Оценка программной продукции. Характеристики качества и руководства по их применению

ГОСТ Р ИСО 9127—94 Системы обработки информации. Документация пользователя и информация на упаковке для потребительских программных пакетов

ИСО/МЭК 6592—2000* Информационная технология. Руководства по документированию прикладных автоматизированных систем

4 Определения

В настоящем стандарте использованы следующие термины с соответствующими определениями:

4.1 **схема классификации** (categorization schema): Упорядоченная комбинация видов и классов, связанных с ПС.

4.2 **вид** (view): Множество соответствующих классов.

4.3 **класс** (category): Конкретно определенный классификационный фрагмент, раздел или группировка ПС, выделенные на основе одного или нескольких атрибутов или характеристик.

5 Сокращения

ПС — программное средство.

6 Концепция классификации ПС

Основной принцип классификации ПС схематически представлен на рисунке 1.

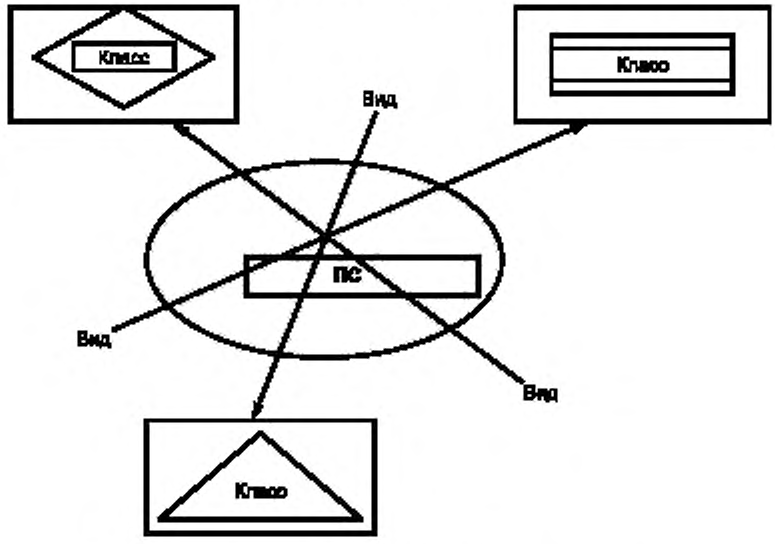

Рисунок 1 — Классификация ПС

Как показано на рисунке 1, классификация ПС комбинируется из ряда видов ПС, а каждый вид содержит классы, соответствующие данному виду. Различные виды рассмотрены в разделе 7. Выбор классов, соответствующих конкретному виду, оставлен на усмотрение пользователя.

Следует отметить, что конкретный класс может присутствовать в нескольких видах, а в ряде случаев область применения одного вида перекрывается областью другого.

* Оригинал международного стандарта — во ВНИИКИ Госстандарта России

6.1 Структура видов

Схема классификации состоит из 16 видов. Данные виды ПС могут быть объединены в следующие группы.

Внутренние виды:

- режим эксплуатации (7.3);
- масштаб ПС (7.4);
- стабильность ПС (7.9);
- функциональные возможности;
- функция ПС (7.1);
- требование защиты (7.13);
- требование надежности (7.14);
- требуемые рабочие характеристики (7.12);
- исходный язык (7.6).

Виды среды:

- прикладная область информационной системы (7.2);
- вычислительная система и среда (7.15);
- класс пользователя (7.8);
- требование к вычислительным ресурсам (7.16);
- критичность ПС (7.7);
- готовность программного продукта (7.10).

Виды данных:

- представление данных (7.5);
- использование программных данных (7.11).

6.2 Выбор видов и классов

В любом конкретном случае, например когда схема классификации применяется к другим стандартам по программной инженерии, использование всех видов может не потребоваться. В таком случае для классификации конкретного ПС может быть достаточен один вид или выбранный набор видов с конкретными классами. Случаи, для которых может потребоваться использование подмножеств видов, охватывают классификацию ПС по отношению к конкретным стандартам программной инженерии, выбор соответствующих стандартов программной инженерии, выбор методов, описание жизненного цикла, определение структуры документа и оценку качества.

При применении схемы классификации к приложениям, для которых необходимо подмножество видов, должны быть использованы все соответствующие виды и связанные с ними классы. Например, если характеризуют ПС по отношению к его среде, может быть пригоден набор видов, составляющих подмножество «Среда» из 6.1. В зависимости от специфики приложения может быть необходимым использование дополнительных видов, таких как требования защиты или надежности.

При некоторых применениях схемы категорирования может быть достаточным использование единственного основного вида. Например критичность ПС может быть основным видом для стандартов по обеспечению качества.

В ряде случаев применения схемы классификации для представления наиболее специфичных характеристик ПС может быть использована комбинация нескольких видов с конкретными классами. Например для определения структуры документации может быть использована функция ПС и класс пользователя.

7 Схема классификации

Схема классификации, связанная с описанием каждого вида, представляет собой перечень классов, соответствующих данному виду. В большинстве случаев такие перечни являются типовыми или открытыми, а не исчерпывающими или полными.

Классы, соответствующие видам, необязательно взаимоисключающи. Для некоторых приложений может быть применен один из классов, а для других — все.

Классы, соответствующие виду, необязательно абстрагированы на одном уровне.

Для пояснения текста настоящего стандарта в нем приведены примеры использования вида. Примеры приведены для иллюстрации и представляют только типовой набор возможных приложений.

Пользователи схемы классификации должны руководствоваться собственными соображениями при выборе соответствующих классов для конкретного приложения или прикладной области.

7.1 Функция ПС

Для вида «функция ПС» классы должны быть определены в зависимости от типа функции, для которой они предназначены.

Примерами классов функции ПС являются:

- обработка деловых сообщений;
- компиляция;
- научные вычисления;
- обработка текстов;
- медицинские системы;
- системы управления.

7.2 Прикладная область информационной системы

Для вида «прикладная область информационной системы» классы должны быть определены в зависимости от типа или класса внешней системы, в которой они устанавливаются.

Например ПС, которое является элементом систем управления процессами, может быть классифицировано как «ПС аппаратуры управления процессом», а ПС, которое является элементом сетевых систем, — как «ПС управления сетью».

Примерами классов прикладной области являются:

- наука;
- бытовые устройства;
- оборудование;
- аппаратура управления процессом;
- предпринимательство;
- система организации сети.

7.3 Режим эксплуатации

Для вида «режим эксплуатации» классы должны быть определены в зависимости от конкретных технологий или типов обработки, принятых в системе программного обеспечения.

Примерами классов режима эксплуатации являются:

- пакетная обработка данных;
- обработка данных в режиме реального времени;
- обработка данных в режиме разделения времени;
- параллельная обработка данных;
- совмещенная обработка данных.

7.4 Масштаб ПС

Для вида «масштаб ПС» классы должны быть определены в зависимости от размера или сложности ПС.

Например размер может быть определен в границах числа строк исходной программы (SLOC), исключая комментарии, и уточнен на уровне языка (то есть в Ассемблере, Фортране, Аде). Сложность может быть определена как функция соответствующего параметра, такого как сложность потока данных. Определения масштаба должны быть нормированы применительно к прикладной области.

Примерами классов масштаба ПС являются:

- малый;
- средний;
- большой.

Следует учитывать, что диапазоны выше названных классов не должны быть жесткими. Напротив, классы должны быть установлены с учетом представления неопределенных или приблизительных диапазонов.

7.5 Представление данных

Для вида «представление данных» классы должны быть определены в зависимости от элементов, типов и структур данных.

Примерами классов представления данных являются:

- последовательный;
- реляционный;
- индексируемый;
- сетевой;
- предметный;
- объектный;
- форматированный файл.

7.6 Исходный язык

Должно быть установлено обозначение типа исходного языка, потому что исходный язык, используемый при разработке ПС, обобщенно представляет или значительно влияет на характеристики ПС.

Примерами классов исходного языка являются:

- традиционный (Кобол, Фортран и т. д.);
- процедурный (Си или эквивалентный);
- функциональный (Лисп или эквивалентный);
- объектно-ориентированный (СИ++ или эквивалентный).

7.7 Критичность ПС

Для вида «критичность ПС» классы должны быть определены в зависимости от оценки уровня целостности продукции, с учетом конкретной методологии оценки и указанием значимости или важности класса. Альтернативно классы могут быть установлены по степени влияния (глобальное, международное и т. д.) или важности для общества (индивидуальное, групповое, деловое и т. д.) повреждений программного обеспечения системы. Повреждение ПС может влиять на безопасность (человеческой жизни, собственности и т. д.) или назначение данного средства (игра, обработка текста, расчеты и т. д.).

Если не использовать оценки уровня целостности, то примерами классов критичности ПС являются:

- национальная безопасность;
- человеческая жизнь;
- социальный хаос или паника;
- организационная безопасность;
- частная собственность;
- секретность.

7.8 Класс пользователя

Для вида «класс пользователя» классы должны быть определены в зависимости от уровня мастерства или характеристик определенного класса пользователей. Пользователем не обязательно является человек.

Примерами классов для вида «класс пользователя» являются:

- начинающий;
- средний;
- специалист (эксперт);
- обычный;
- случайный;
- другая система программного обеспечения;
- технические средства.

7.9 Стабильность ПС

ПС должно быть классифицировано в зависимости от его внутренних эволюционных аспектов или стабильности в терминах характеристик системы, частью которой оно является.

Примерами классов стабильности ПС являются:

- постоянное внесение изменений;
- дискретное внесение изменений;
- маловероятное внесение изменений.

7.10 Готовность программного продукта

Для вида «готовность программного продукта» классы должны быть определены в зависимости от типа (ов) имеющегося в наличии ПС.

Примерами классов готовности ПС являются:

- наличие в готовом виде;
- заказное;
- общедоступное;
- запатентованное (оригинальная разработка).

7.11 Использование программных данных

Для вида «использование программных данных» классы должны быть определены в зависимости от типа применения программных данных.

Примерами классов использования программных данных являются:

- для единственного пользователя (индивидуальные);
- для множества пользователей;
- с конкурентным взаимоисключением.

7.12 Требуемые рабочие характеристики

Для вида «требуемые рабочие характеристики» классы должны быть определены в зависимости от производительности ПС в терминах «емкости», «производительности» или «длительности обработки», при этом каждый класс должен быть оценен по степени или уровню.

Примерами классов требуемых рабочих характеристик являются:

- емкость:
  - высокая,
  - средняя,
  - низкая;
- длительность обработки (время отклика):
  - быстрая,
  - умеренная,
  - медленная;
- производительность:
  - большая,
  - средняя,
  - малая.

7.13 Требование защиты

Для вида «требование защиты» ПС должны быть классифицированы по уровню защиты от несанкционированного доступа, контрольному следу и обеспечению живучести (устойчивости к внешним воздействиям). Могут быть установлены дополнительные классы требований защиты.

Примеры классов для требования защиты показаны в таблице на рисунке 2.

	Сильная	Средняя	Слабая
Защита от несанкционированного доступа			
Контрольный след			
Защита программ и данных			

Рисунок 2 — Классы требования защиты

7.14 Требование надежности

Для вида «требование надежности» ПС должны быть классифицированы по уровню требуемой надежности, включая завершенность, отказоустойчивость и восстанавливаемость.

Примеры классов для требования надежности показаны в таблице на рисунке 3.

	Высокая	Средняя	Низкая
Завершенность			
Отказоустойчивость			
Восстанавливаемость			

Рисунок 3 — Классы требования надежности

7.15 Вычислительная система и среда

Для вида «вычислительная система и среда» ПС должно быть идентифицированы с точки зрения его использования в конкретной целевой вычислительной системе.

Примерами классов вычислительной системы и среды являются:

- микропроцессорное управление (включая рабочие станции, а также персональные, переносимые и портативные компьютеры);
- универсальные компьютеры;
- специализированное микропрограммирование;
- нефоннеймановская машина;
- операционная система;
- система реального времени.

7.16 Требование к вычислительным ресурсам

Для вида «требование к вычислительным ресурсам» ПС должно быть идентифицировано по отношению к требованиям, определяемым компьютером. Требования могут быть установлены в границах необходимого количества ресурсов.

Примерами классов вычислительных ресурсов являются:

- требования к центральному обрабатывающему устройству (процессору);
- требования к оперативной (основной) памяти;
- требования к внешней памяти;
- требования к памяти на дисках;
- требования к локальной вычислительной сети.

8 Применение схемы

В настоящем разделе приведены примеры применения схемы классификации ПС для некоторых возможных прикладных областей.

8.1 Применение к области действия стандартов

Стандарты по документированию могут быть разработаны для ПС конкретных масштабов, таких как большие прикладные системы (ИСО/МЭК 6592), и для конкретных признаков готовности ПС, таких как пакетное ПС (ГОСТ Р ИСО 9127).

8.2 Применение к стандартам

В некоторых случаях конкретное содержание стандарта определяет или использует конкретные виды или классы ПС. Для иллюстрации этого таблица на рисунке 4 представляет связь шести характеристик качества ПС из ГОСТ Р ИСО/МЭК 9126 с видами из схемы классификации. В таблице взаимосвязи определены как основные (О) или второстепенные (В).

Характеристика качества по ГОСТ Р ИСО/МЭК 9126

Классификация видов ПС	Функциональные возможности	Надежность	Практичность	Эффективность	Сопровождаемость	Мобильность
Функция ПС	О					
Режим эксплуатации	О					
Прикладная область информационной системы	О		В			
Масштаб ПС				О	О	В
Представление данных				О		В
Критичность ПС	О	В				
Класс пользователя			О			
Требуемые рабочие характеристики				О		
Стабильность ПС					О	В
Требование защиты	О					
Требование надежности		О				
Вычислительная система и среда	В					В
Требование к вычислительным ресурсам				В		В
Готовность программного продукта	В				О	В
Использование программных данных	О	В				
Исходный язык	О			В		О

Рисунок 4 — Отображение ГОСТ Р ИСО/МЭК 9126 на схему классификации

8.3 Применение к программным пакетам

Схема классификации в целом применима для классификации конкретных программных пакетов. Следующий пример показывает классификацию гипотетического программного пакета для обработки текста. Представленное исходное описание обеспечивает достаточную информацию для классификации пакета и содержит обоснования его применения в действующей системе.

8.4 Пример классификации. Пакет для обработки текста

Функция ПС

Текстовый процессор.

Режим эксплуатации

Интерактивная обработка (пользователь ПС вводит текст или команды, которые обрабатываются ПС).

Прикладная область информационной системы

Предметная область делового и персонального применения (предназначена для делового применения, но программа полезна для персональных или других приложений).

Масштаб ПС

Малый и средний (меньший, в SLOC, по сравнению с подобными пакетами;
расширение возможностей несколько увеличивает сложность).

Представление данных

Объектные (для командной операции) и форматированный файл (для командной операции, текста ввода — вывода и хранения данных).

Критичность ПС

Экономично (по времени) и удобно для индивидуального применения.

Класс пользователя

Средний (пользователь должен иметь опыт в обработке текста на пишущей машине, но может не иметь навыков в использовании программ).

Требуемые рабочие характеристики

Емкость — средняя и высокая (размер документа ограничен только доступностью вычислительных ресурсов).

Время отклика — быстрое (принимает более 50 символов / с при вводе данных с быстрой регенерацией экрана).

Производительность — малая (одновременно один документ).

Стабильность ПС

С контролируемым изменением (обычная модернизация и новые выпуски).

Требование защиты

	Сильная	Средняя	Слабая
Защита от несанкционированного доступа		—	
Контрольный след			—
Защита программ и данных	—		

(доступно использование пароля и шифрация данных; контрольный след не обеспечивается; низкая интенсивность отказов).

Требование надежности

	Высокая	Средняя	Низкая
Завершенность	—		
Отказоустойчивость	—		
Восстанавливаемость	—		

(пакет ранее проходил ряд переделок и модернизаций; возможны новые ошибки; редактирование прерывается с клавиатуры).

Вычислительная система и среда

Управляемый микропроцессор — персональный / переносной / портативный (конкретные типы, модели и классы компьютера должны быть определены).

Требования к вычислительным ресурсам

Оперативная память — память с произвольной выборкой (должны быть установлены минимальный и рекомендуемый объемы).

Внешняя память — массовое запоминающее устройство (должны быть установлены минимальные и рекомендуемые объемы и типы).

Готовность программного продукта

Готовый коммерческий; запатентованный (лицензионный).

Использование программных данных

В обычном случае — единственный пользователь, но возможна очередь пользователей без снижения эффективности.

Исходный язык

Си++.

УДК 681.3.06 : 006.354 ОКС 35.080 П 85 ОКСТУ 5001

Ключевые слова: обработка данных, вычислительные машины, программные средства вычислительных машин, виды, классы, описание

Редактор *В.П. Огурцов*
Технический редактор *Л.А. Гусева*
Корректор *И.Л. Рыбалко*
Компьютерная верстка *И.А. Налейкиной*

Изд. лиц. № 02354 от 14.07.2000. Сдано в набор 22.07.2002. Подписано в печать 09.08.2002. Усл. печ. л. 1,86. Уч.-изд.л. 1,20.
Тираж 704 экз. С 7029. Зак. 664.

ИПК Издательство стандартов, 107076 Москва, Колодезный пер., 14.
http://www.standards.ru e-mail. info@standards.ru
Набрано в Издательстве на ПЭВМ
Филиал ИПК Издательство стандартов — тип. «Московский печатник», 103062 Москва, Лялин пер., 6.
Плр № 080102

к ГОСТ Р ИСО/МЭК ТО 12182—2002 Информационная технология. Классификация программных средств

В каком месте	Напечатано	Должно быть
С. 1. Дата введения	2007—07—01	2003—07—01

(ИУС № 12 2002 г.)

gost.e-ecolog.ru