

## Оглавление

Лекция 4 Документация в процессе проектирования ПО .....	2
4.1 Программный продукт и его артефакты.....	2
4.2 Программный проект.....	4
4.3 Процесс проектирования .....	4
4.4 Этап внедрения.....	5
4.5 Методы и модели программной инженерии .....	5
4.6 Документация, используемая в процессе разработки программных средств.....	7

## Лекция 4 Документация в процессе проектирования ПО

Ниже приводятся лишь некоторые основные понятия программной инженерии, имеющие отношения к наиболее общим («верхним») уровням проектирования ПО на разных этапах проектирования.

### 4.1 Программный продукт и его артефакты

При разработке программного обеспечения под **программным продуктом** понимают совокупность созданного программного приложения и все его артефакты<sup>1</sup>, появившиеся при работе над этим приложением.

**Программный продукт** – результат реализации программного проекта, обладающий заявленной функциональностью и потребительскими характеристиками.

**Артефакт программного продукта** – неотъемлемая часть результата и процесса выполнения программного проекта, реализованная в виде документации, программного кода (исходного или скомпилированного) или его части (например, модуля).

Примеры артефактов, соответствующих составляющих приложения приведены в табл.4.1.

Таблица 4.1

Артефакты программного продукта

Составляющие программного продукта и процесса	Артефакты
Требования	Спецификация требований к программному продукту
Программная архитектура	Проектная модель
Детальная проектирование	Исходный и объектный код

---

<sup>1</sup> **Артефакт** (лат. *artefactum* от *arte* — искусственно + *factus* — сделанный) в обычном понимании — любой искусственно созданный объект, продукт человеческой деятельности.

Реализация	
Тестирование	Тестовые процедуры и тестовые варианты
Внедрение и сопровождение	Документация по программному обеспечению

Выделяют две категории программных продуктов.

1. Общего назначения (коробочное программное обеспечение), создаваемое для распространения на открытом рынке;
2. Программные продукты, выполненные по заказу конкретного потребителя или целевой категории. Они обладают узконаправленной функциональностью.

**Правовые аспекты защиты программных продуктов** – авторское право, патентная защита, закон о производственных секретах, лицензионное соглашение и контракты.

**Качество программного обеспечения (программного продукта)** – совокупность наиболее важных характеристик программного продукта (например, надёжность), а также характеристик самого процесса разработки (например, количество дефектов на тысячу строк кода), измеряемых количественно по тем или иным методикам, на основе которых можно сделать заключение о соответствии продукта и/или процесса заранее определённым показателям.

**Прототип** – программный продукт, по ряду ключевых на данный момент характеристик близкий к разрабатываемому. Прототип предназначен для демонстрации для демонстрации результатов заказчику с целью определения его мнения относительно интерфейса или части реализованной функциональности. Как правило прототипы используют для своевременной корректировки

требований к программному обеспечению к программному продукту в процессе разработки.

## 4.2 Программный проект

**Проект** – протяжённое во времени предприятие, направленное на создание уникальных продуктов, услуг или достижение иных результатов.

**IT-проект** (проект с позиций программной инженерии) – это совокупность действий, необходимая для создания артефактов программного продукта. Любой проект включает в себя взаимодействие с заказчиком, написание документации, кодирование и тестирование. А также – процесс реализации комплекса мероприятий, направленных на создание программного продукта.

**Требования к программному обеспечению** – документ, отражающий, что должно делать разрабатываемое программное обеспечение.

**Исходный код** – программный код, написанный на каком-либо языке программирования.

**Объектный код** – переведённый на машинный язык исходный код, ещё не подверженный особой упаковке, характерной для конкретной операционной системы.

**Двоичный код** – переведённый на машинный язык, понятный ЭВМ исходный код программы.

## 4.3 Процесс проектирования

**Архитектура программного обеспечения** – модель ПО на самом высоком уровне, формализованная теми или иными средствами

**Проектирование программного обеспечения** – создание модели программного обеспечения с применением тех или иных средств, языков и стандартов, уровень детализации которой находится между архитектурой программного обеспечения и потребительскими характеристиками.

**Детальное проектирование** - создание подробной модели разрабатываемой программной системы с применением псевдокода, блок-схем и др., достаточной для написания программного кода.

**Инспектирование** – коллективное исследование артефактов проекта, направленное на выявление дефектов, осуществляемое, как правило, лицами (разработчиками или их группами), не участвующими в создании inspectируемых артефактов.

**Тестирование** – комплекс мероприятий, направленный на выявление дефектов в готовом программном обеспечении и/или его составляющих.

Управление конфигурациями программного обеспечения – поддержание соответствия версий всех артефактов создаваемого программного продукта в процессе разработки.

#### **4.4 Этап внедрения**

**Сопровождение программного обеспечения** – комплекс мероприятий, направленный на выявление дефектов в готовом программном обеспечении и/или его составляющих.

#### **4.5 Методы и модели программной инженерии**

Метод программной инженерии — это структурный подход к созданию ПО, который способствует производству высококачественного продукта эффективным в экономическом аспекте способом. В этом определении есть две основные составляющие: (а) создание высококачественного продукта и (б) экономически эффективным способом. Иными словами, метод – это то, что обеспечивает решение основной задачи программной инженерии: создание качественного продукта при заданных ресурсах времени, бюджета, оборудования, людей.

Начиная с 70-х годов создано достаточно много методов разработки ПО. Наиболее известны:

- Метод структурного анализа и проектирования Том ДеМарко (1978),
- Метод сущность-связь проектирования информационных систем Чен (1976)
- Метод объектно-ориентированного анализа Буч (1994), Рамбо (1991).

**Метод программной индустрии основан на идее создания моделей ПО с поэтапным преобразованием этих моделей в программу – окончательную модель решаемой задачи.**

Так, на этапе спецификаций создается **модель** – описание требований, которая далее преобразуется в модель проекта ПО, проект – в программный код. При этом важно, чтобы модели метода представлялись графически с помощью некоторого языка представления моделей.

**Методы должны включать в себя следующие компоненты:**

- Описание моделей системы и нотация, используемая для описания этих моделей (например, объектные модели, конечно-автоматные модели и т.д.)
- Правила и ограничения, которые надо выполнять при разработке моделей (например, каждый объект должен иметь одинаковое имя)
- Рекомендации — **эвристики**<sup>2</sup>, характеризующие хорошие приемы проектирования в данном методе (скажем, рекомендация о том, что ни у **одного объекта не должно быть больше семи подобъектов**)
- Руководство по применению метода — описание последовательности работ (действий), которые надо выполнить для построения моделей (все атрибуты должны быть задокументированы до определения операций, связанных с этим объектом)

Нет идеальных методов, все они применимы только для тех или иных случаев.

---

<sup>2</sup> **Эвристика** (от древнегреческого *eurísko* – «отыскиваю», «открываю») – совокупность логических приемов, методов и правил, облегчающих и упрощающих решение познавательных, конструктивных, практических задач. Эвристика – это момент открытия нового, а также методы, которые используются в процессе этого открытия. Эвристикой еще называют науку, которая имеет дело с изучением творческой деятельности. В педагогике под этой категорией подразумевается метод обучения.

Нет абсолютных методов – применяемые на практике методы могут включать элементы различных подходов. **Выбор метода составляет задачу специалиста по программной инженерии.**

**Модель в программной инженерии** – отображение в той или иной нотации (форме), понятной квалифицированному специалисту, задачи и результата того или иного этапа процесса разработки программного проекта. Например, различают следующие модели:

- Модель прецедентов (требований);
- Модель классов;
- Модель сущность-связь.

#### **4.6 Документация, используемая в процессе разработки программных средств**

Инициация проекта ПС, процессы разработки, внедрения и сопровождения должны сопровождаться созданием соответствующего документационного сопровождения. Это сложное и трудоёмкое дело определяется соответствующими стандартами.

Ниже представлены лишь некоторые базовые аспекты. Содержание некоторых документов будет подробнее рассмотрено в одной из следующих лекций.

Достаточно полное описание можно найти по адресу:

[http://www.tinlib.ru/kompyutery\\_i\\_internet/programmnoe\\_obespechenie\\_vstroennyh\\_sistem\\_obshie\\_trebovaniya\\_k\\_razrabotke\\_i\\_dokumentirovaniyu/p12.php](http://www.tinlib.ru/kompyutery_i_internet/programmnoe_obespechenie_vstroennyh_sistem_obshie_trebovaniya_k_razrabotke_i_dokumentirovaniyu/p12.php) ,

представляющей материалы on-line библиотеки. Там же можно найти Государственный стандарт России , подготовленный на основе применения аутентичного текста технических рекомендаций ИСО/МЭК ТО 9294-90 «Информационная

технология. Руководство по управлению документированием программного обеспечения»<sup>3</sup>

#### ***4.6.1 Назначение, классификация и роль документации в процессе проектировании программного средства***

Данная тема очень обширна важна. Ниже приводим базовые понятия<sup>4</sup>, которые основываются в свою очередь на следующих стандартах:

1. ANSI/IEEE Std 1063-1988, IEEE Standard for Software User Documentation.
2. ANSI/IEEE Std 830-1984, IEEE Guide for Software Requirements Specification.
3. ANSI/IEEE Std 1016-1987, IEEE Recommended Practice for Software Design Description.
4. ANSI/IEEE Std 1008-1987, IEEE Standard for Software Unit Testing.
5. ANSI/IEEE Std 1012-1986, IEEE Standard for Software Verification and Validation Plans.
7. ANSI/IEEE Std 983-1986, IEEE Guide for Software Quality Assurance Planning.
8. ANSI/IEEE Std 829-1983, IEEE Standard for Software Test Documentation

При разработке ПС создается и используется большой объем разнообразной документации. Она необходима как:

- средство передачи информации между разработчиками ПС,
- как средство управления разработкой ПС и
- как средство передачи пользователям информации, необходимой для применения и сопровождения ПС.

#### **На создание этой документации приходится большая доля стоимости ПС.**

Эту документацию можно разбить на две группы:

- Документы управления разработкой ПС.

---

3

[http://www.tinlib.ru/kompyutery\\_i\\_internet/informacionnaja\\_tehnologija\\_rukovodstvo\\_po\\_upravleniyu\\_dokumentirovaniem\\_programmnogo\\_obespechenija/index.php](http://www.tinlib.ru/kompyutery_i_internet/informacionnaja_tehnologija_rukovodstvo_po_upravleniyu_dokumentirovaniem_programmnogo_obespechenija/index.php)

4

<http://bourabai.ru/alg/pro13.htm>



- Документы, входящие в состав ПС.

**Документы управления разработкой ПС (software process documentation)**

управляют и протоколируют процессы разработки и сопровождения ПС, обеспечивая связи внутри коллектива разработчиков ПС и между коллективом разработчиков и *менеджерами ПС (software managers)* - лицами, управляющими разработкой ПС. Эти документы могут быть следующих типов:

- **Планы, оценки, расписания.** Эти документы создаются менеджерами для прогнозирования и управления процессами разработки и сопровождения ПС.
- **Отчеты об использовании ресурсов в процессе разработки.** Создаются менеджерами.
- **Стандарты.** Эти документы предписывают разработчикам, каким принципам, правилам, соглашениям они должны следовать в процессе разработки ПС. Эти стандарты могут быть как международными или национальными, так и специально созданными для организации, в которой ведется разработка ПС.
- **Рабочие документы.** Это основные технические документы, обеспечивающие связь между разработчиками. Они содержат фиксацию идей и проблем, возникающих в процессе разработки, описание используемых стратегий и подходов, а также рабочие (временные) версии документов, которые должны войти в ПС.
- **Заметки и переписка.** Эти документы фиксируют различные детали взаимодействия между менеджерами и разработчиками.

**Документы, входящие в состав ПС (software product documentation)**, описывают программы ПС как с точки зрения их применения пользователями, так и с точки зрения их разработчиков и сопроводителей (в соответствии с назначением ПС). Здесь следует отметить, что эти документы будут использоваться не только на стадии эксплуатации ПС (в ее фазах применения и сопровождения), но и на стадии разработки для управления процессом разработки (вместе с рабочими документами) - во всяком случае, **они должны быть проверены**

(протестированы) на соответствие программам ПС. Эти документы образуют два комплекта с разным назначением:

- Пользовательская документация ПС (П-документация).
- Документация по сопровождению ПС (С-документация).

#### **4.6.2 Пользовательская документация программных средств**

**Пользовательская документация ПС (user documentation)** объясняет пользователям, как они должны действовать, чтобы применить разрабатываемое ПС. Она необходима, если ПС предполагает какое-либо взаимодействие с пользователями. К такой документации относятся документы, которыми должен руководствоваться пользователь при *инсталляции* ПС (при установке ПС с соответствующей настройкой на среду применения ПС), при применении ПС для решения своих задач и при управлении ПС (например, когда разрабатываемое ПС будет взаимодействовать с другими системами). Эти документы частично затрагивают вопросы сопровождения ПС, но не касаются вопросов, связанных с модификацией программ.

В связи с этим следует различать две категории пользователей ПС: **ординарных** пользователей ПС и **администраторов** ПС.

**Ординарный пользователь ПС (end-user)** использует ПС для решения своих задач (в своей предметной области). Это может быть инженер, проектирующий техническое устройство, или кассир, продающий железнодорожные билеты с помощью ПС. Он может и не знать многих деталей работы компьютера или принципов программирования.

**Администратор ПС (system administrator)** управляет использованием ПС ординарными пользователями и осуществляет сопровождение ПС, не связанное с модификацией программ. Например, он может регулировать права доступа к ПС между ординарными пользователями, поддерживать связь с поставщиками ПС или выполнять определенные действия, чтобы поддерживать ПС в рабочем состоянии, если оно включено как часть в другую систему.

Состав пользовательской документации **зависит от аудиторий пользователей**, на которые ориентировано разрабатываемое ПС, и от режима использования документов. Под *аудиторией* здесь понимается контингент пользователей ПС, у которого есть необходимость в определенной пользовательской документации ПС. Удачный пользовательский документ существенно зависит от точного определения аудитории, для которой он предназначен. Пользовательская документация должна содержать информацию, необходимую для каждой аудитории.

Под **режимом использования** документа понимается способ, определяющий, каким образом используется этот документ. Обычно пользователю достаточно больших программных систем требуются либо документы для изучения ПС (использование в виде *инструкции*), либо для уточнения некоторой информации (использование в виде *справочника*).

Типичным можно считать следующий состав пользовательской документации для достаточно больших ПС:

- **Общее функциональное описание ПС.** Дает краткую характеристику функциональных возможностей ПС. Предназначено для пользователей, которые должны решить, насколько необходимо им данное ПС.
- **Руководство по инсталляции ПС.** Предназначено для администраторов ПС. Оно должно детально предписывать, как устанавливать системы в конкретной среде, в частности, должно содержать описание компьютерно-читываемого носителя, на котором поставляется ПС, файлы, представляющие ПС, и требования к минимальной конфигурации аппаратуры.
- **Инструкция по применению ПС.** Предназначена для ординарных пользователей. Содержит необходимую информацию по применению ПС, организованную в форме удобной для ее изучения.
- **Справочник по применению ПС.** Предназначен для ординарных пользователей. Содержит необходимую информацию по применению ПС, организованную в форме удобной для избирательного поиска отдельных деталей.

- ***Руководство по управлению ПС.*** Предназначено для администраторов ПС. Оно должно описывать сообщения, генерируемые, когда ПС взаимодействует с другими системами, и как должен реагировать администратор на эти сообщения. Кроме того, если ПС использует системную аппаратуру, этот документ может объяснять, как сопровождать эту аппаратуру.

**Разработка пользовательской документации начинается сразу после создания внешнего описания.** Качество этой документации может существенно определять успех ПС. Она должна быть достаточно проста и удобна для пользователя (в противном случае это ПС, вообще, не стоило создавать). Поэтому, хотя черновые варианты (наброски) пользовательских документов создаются основными разработчиками ПС, к созданию их окончательных вариантов часто привлекаются **профессиональные технические писатели.** Кроме того, для обеспечения качества пользовательской документации разработан ряд стандартов, в которых предписывается порядок разработки этой документации, формулируются требования к каждому виду пользовательских документов и определяются их структура и содержание.

#### ***4.6.3 Документация по сопровождению программных средств***

***Документация по сопровождению ПС (system documentation)*** описывает ПС с точки зрения ее разработки. Эта документация необходима, если ПС предполагает изучение того, как оно устроена (сконструирована), и модернизацию его программ.

**Сопровождение** - это продолжающаяся разработка. Поэтому в случае необходимости модернизации ПС к этой работе привлекается специальная команда разработчиков-сопроводителей.

Этой команде придется иметь дело с такой же документацией, которая определяла деятельность команды первоначальных (основных) разработчиков ПС, - с той лишь разницей, что эта документация для команды разработчиков - сопроводителей будет, как правило, чужой (она создавалась другой коман-

дой). Чтобы понять строение и процесс разработки модернизируемого ПС, команда разработчиков-сопроводителей должна изучить эту документацию, а затем внести в нее необходимые изменения, повторяя в значительной степени технологические процессы, с помощью которых создавалось первоначальное ПС.

**Документация по сопровождению ПС можно разбить на две группы:**

1. документация, определяющая строение программ и структур данных ПС и технологию их разработки;
2. документацию, помогающую вносить изменения в ПС.

Документация первой группы содержит итоговые документы каждого технологического этапа разработки ПС. Она включает следующие документы:

- Внешнее описание ПС (Requirements document).
- Описание архитектуры ПС (description of the system architecture), включая внешнюю спецификацию каждой ее программы (подсистемы).
- Для каждой программы ПС - описание ее модульной структуры, включая внешнюю спецификацию каждого включенного в нее модуля.
- Для каждого модуля - его спецификация и описание его строения (design description).
- Тексты модулей на выбранном языке программирования (program source code listings).
- Документы установления достоверности ПС (validation documents), описывающие, как устанавливалась достоверность каждой программы ПС и как информация об установлении достоверности связывалась с требованиями к ПС.

Документы установления достоверности ПС включают, прежде всего, документацию по тестированию (схема тестирования и описание комплекта тестов), но могут включать и результаты других видов проверки ПС, например, доказательства свойств программ. **Для обеспечения приемлемого качества этой документации полезно следовать общепринятым рекомендациям и стандартам.**

**Документация второй группы содержит:**

- ***Руководство по сопровождению ПС*** (system maintenance guide), которое описывает особенности реализации ПС (в частности, трудности, которые пришлось преодолевать) и как учтены возможности развития ПС в его строении (конструкции). В нем также фиксируются, какие части ПС являются аппаратно- и программно-зависимыми.

Общая проблема сопровождения ПС - обеспечить, чтобы все его представления шли в ногу (оставались согласованными), когда ПС изменяется. Чтобы этому помочь, связи и зависимости между документами и их частями должны быть отражены в руководстве по сопровождению, и зафиксированы в базе данных управления конфигурацией.

#### ***4.6.3 Перечень документов программного проекта согласно руководству РМВОК***

Руководство к своду знаний по управлению проектами содержит перечень документов, которые должны создаваться в процессе его реализации на разных стадиях. Естественно, что этими рекомендациями можно и следует руководствоваться и при реализации программного проекта.

##### **4.6.3.1 Классификация проектных документов**

Проектные документы делятся на три группы:

**1-я группа** – документы непосредственно по управлению проектом (Устав проекта, паспорт проекта, календарный план проекта, ресурсный план проекта, отчет о статусе проекта и т.д.).

**2-я группа** – документы, разрабатываемые на стадиях создания информационных систем (техническое задание, программа и методика испытаний, инструкции к системе и т.д.).

**3-я группа** – финансовые, договорные, организационно-распорядительные документы (договор, акты сдачи-приемки работ, счета, счета-фактуры, приказ о начале/завершении проекта, акт о вводе в промышленную эксплуатацию и т.д.).

#### **4.6.3.2 Основные документы, разрабатываемые на стадиях создания ИС**

**1.Техническое задание.** В нем фиксируется назначение и цели создания системы, характеристика объекта внедрения, требования к системе, перечень разрабатываемых на ИС документов и т.д. Значение этого документа в проекте сложно переоценить.

**2.Эскизный проект** - совокупность конструкторских документов, которые должны содержать принципиальные конструктивные решения, дающие общее представление об устройстве и принципе работы изделия, а также данные, определяющие назначение, основные параметры и габаритные размеры разрабатываемого изделия.

**3.Технический проект** - совокупность конструкторских документов, которые должны содержать окончательные технические решения, дающие полное представление об устройстве разрабатываемого изделия, и исходные данные для разработки рабочей документации.

**4.План-график работ.** Документ устанавливает перечень работ, сроки выполнения и исполнителей работ, связанных с созданием АС. Документ для каждой работы, включенной в перечень, содержит:

- наименование работы;
- дату начала и окончания работы;
- наименование подразделения - участника работы;
- фамилию и должность ответственного исполнителя;
- форму представления результатов работы.

**5.Программа и методика испытаний.** Программа и методика испытаний устанавливает необходимый и достаточный объем испытаний, обеспечивающий заданную достоверность получаемых результатов.

**6.Акт приемки в опытную эксплуатацию.**

**7.Приказ о начале опытной эксплуатации (ее частей).** Документ содержит:

- наименование автоматизированной (информационной) системы в целом или ее частей, проходящей опытную эксплуатацию;

- наименование организации-разработчика,
- организаций-соисполнителей;
- сроки проведения опытной эксплуатации; список должностных лиц организации-заказчика и организации-разработчика, ответственных за проведение опытной эксплуатации;
- перечень подразделений организации-заказчика, участвующих в проведении опытной эксплуатации.

#### **8.Приказ о составе приемочной комиссии.** Документ содержит:

- наименование принимаемой АС в целом или ее частей;
- сведения о составе комиссии;
- основание для организации комиссии;
- наименование организации-заказчика;
- наименование организации-разработчика, организаций-соисполнителей;
- назначение и цели работы комиссии;
- сроки начала и завершения работы комиссии;
- указание о форме завершения работы комиссии.

#### **9.Протокол испытаний.** Документ содержит:

- наименование объекта испытаний;
- список должностных лиц, проводивших испытания;
- цель испытаний;
- сведения о продолжительности испытаний; перечень пунктов технического задания на создание АС, на соответствие которым проведены испытания;
- перечень пунктов "Программы испытаний", по которым проведены испытания;
- сведения о результатах наблюдений за правильностью функционирования АС;
- сведения об отказах, сбоях и аварийных ситуациях, возникающих при испытаниях;



- сведения о корректировках параметров объекта испытания и технической документации.
- **10.Приказ о вводе в промышленную эксплуатацию.** Документ содержит:
  - состав функций АС или ее частей, технических и программных средств, принимаемых в промышленную эксплуатацию;
  - список должностных лиц и перечень подразделений организации-заказчика, ответственных за работу АС;
  - порядок и сроки введения новых форм документов (при необходимости);
  - порядок и сроки перевода персонала на работу в условиях функционирования АС.

#### **11.Руководство по эксплуатации ИС.**

**12.Руководство пользователей ИС** (в соответствии с функциональными ролями).

**13.Статус-отчеты.** Статус-отчетов может быть несколько: отчет, предоставляемый вам подрядчиком; отчет, который готовите непосредственно вы для коллегиальных органов, контролирующих ход проекта. Статус-отчеты регулярно создаются на протяжении всего проекта с определенной периодичностью. Причем, периодичность предоставления отчета подрядчиком определяете вы сами (как правило, 1 раз в неделю), а периодичность предоставления отчета для коллегиальных органов определена либо Уставом проекта, либо локальными нормативными документами по управлению проектами.

