

# Глава 5: Обслуживание программного обеспечения

Из SWEBOK

## Содержание

- 1 Основы обслуживания программного обеспечения
  - 1.1 Определения и терминология
  - 1.2 Характер технического обслуживания
  - 1.3 Необходимость технического обслуживания
  - 1.4 Большая часть расходов на техническое обслуживание
  - 1.5 Эволюция программного обеспечения
  - 1.6 Категории обслуживания
- 2 ключевые проблемы в обслуживании программного обеспечения
  - 2.1 Технические проблемы
  - 2.2 Вопросы управления
  - 2.3 Оценка стоимости обслуживания
  - 2.4 Измерение обслуживания программного обеспечения
- 3 Процесс технического обслуживания
  - 3.1 Процессы технического обслуживания
  - 3.2 Работы по техническому обслуживанию
- 4 техники ухода
  - 4.1 Понимание программы
  - 4.2 Реинжиниринг
  - 4.3 Обратный инжиниринг
  - 4.4 Миграция
  - 4.5 Выход на пенсию
- 5 инструментов обслуживания программного обеспечения

## АКРОНИМЫ

<b>Г-Н</b>	Запрос на изменение
<b>пиар</b>	Отчет о проблеме
<b>СКМ</b>	Управление конфигурацией программного обеспечения
<b>Соглашение об уровне обслуживания</b>	Соглашение об уровне обслуживания
<b>SQA</b>	Гарантия качества программного обеспечения
<b>ВиВ</b>	Верификация и валидация

## ВВЕДЕНИЕ

Усилия по разработке программного обеспечения приводят к выпуску программного продукта, удовлетворяющего требованиям пользователя. Соответственно, программный продукт должен изменяться или развиваться. После запуска выявляются дефекты, меняются операционные среды и

появляются новые требования пользователей. Фаза технического обслуживания в жизненном цикле начинается после гарантийного периода или предоставления поддержки после внедрения, но действия по техническому обслуживанию выполняются намного раньше.

Сопровождение программного обеспечения является неотъемлемой частью жизненного цикла программного обеспечения. Однако ему не уделялось такого же внимания, как другим этапам. Исторически сложилось так, что разработка программного обеспечения в большинстве организаций имела гораздо более высокий профиль, чем его обслуживание. Теперь это меняется, поскольку организации стремятся выжать максимальную отдачу от своих инвестиций в разработку программного обеспечения, сохраняя программное обеспечение в рабочем состоянии как можно дольше. Парадигма открытого исходного кода привлекла дополнительное внимание к проблеме поддержки программных артефактов, разработанных другими.

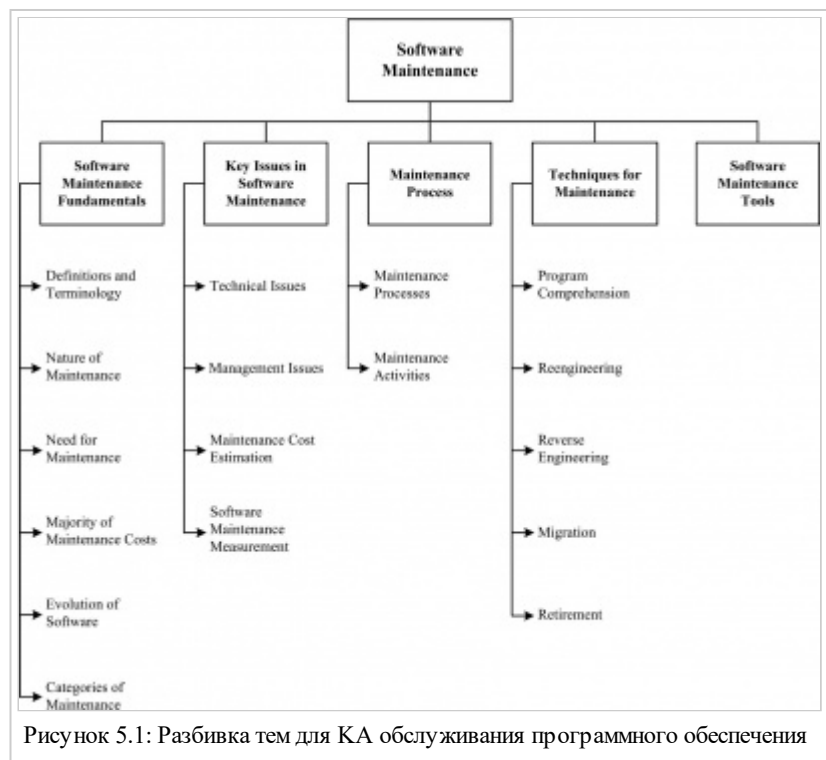
В данном *Руководстве* обслуживание программного обеспечения определяется как совокупность действий, необходимых для обеспечения рентабельной поддержки программного обеспечения. Действия выполняются на этапе до поставки, а также на этапе после доставки. Действия перед поставкой включают планирование операций после поставки, ремонтпригодность и определение логистики для действий по переходу [1\*, с6с9]. Действия после поставки включают в себя модификацию программного обеспечения, обучение и работу со службой поддержки или взаимодействие со службой поддержки.

Область знаний по сопровождению программного обеспечения (КА) связана со всеми другими аспектами разработки программного обеспечения. Таким образом, это описание КА связано со всеми другими КА разработки программного обеспечения в *Руководстве*.

## РАЗБИВКА ТЕМ ПО СОПРОВОЖДЕНИЮ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Разбивка тем для КА обслуживания программного обеспечения показана на рисунке 5.1.

### 1 Основы обслуживания программного обеспечения



В этом первом разделе представлены концепции и терминология, которые формируют основу для понимания роли и объема обслуживания программного обеспечения. Темы содержат определения и подчеркивают, почему необходимо техническое обслуживание. Категории обслуживания программного обеспечения имеют решающее значение для понимания его основного значения.

#### 1.1 Определения и терминология

[ 1 , с3] [ 2 , с1с2, с2с2]

Цель сопровождения программного обеспечения определена в международном стандарте сопровождения программного обеспечения: ISO/IEC/IEEE 14764 [1\*].1 В контексте разработки программного обеспечения сопровождение программного обеспечения, по сути, является одним из многих технических процессов.

Целью обслуживания программного обеспечения является модификация существующего программного обеспечения с сохранением его целостности. В международном стандарте также говорится о важности некоторых действий по обслуживанию до окончательной поставки программного обеспечения (предварительные действия). Примечательно, что в стандарте IEEE 14764 подчеркивается важность аспектов обслуживания перед поставкой — например, планирования.

### 1.2 Характер технического обслуживания

[ 2 , c1c3]

Сопровождение программного обеспечения поддерживает программный продукт на протяжении всего его жизненного цикла (от разработки до эксплуатации). Запросы на модификации регистрируются и отслеживаются, определяется влияние предлагаемых изменений, модифицируются код и другие программные артефакты, проводится тестирование и выпускается новая версия программного продукта. Также пользователям предоставляется обучение и ежедневная поддержка. Термин «мейнтейнер» определяется как организация, выполняющая работы по техническому обслуживанию. В этом КА этот термин иногда будет относиться к лицам, которые выполняют эти действия, противопоставляя их разработчикам.

IEEE 14764 определяет основные виды деятельности по сопровождению программного обеспечения, такие как реализация процесса, анализ проблем и модификаций, внедрение модификаций, проверка/принятие сопровождения, миграция и вывод из эксплуатации. Эти действия обсуждаются в разделе 3.2, Действия по техническому обслуживанию.

Специалисты по сопровождению могут учиться на знаниях разработчиков о программном обеспечении. Контакт с разработчиками и раннее участие сопровождающего помогает сократить общие усилия по обслуживанию. В некоторых случаях с первоначальным разработчиком невозможно связаться или он перешел к другим задачам, что создает дополнительную проблему для сопровождающих. Сопровождение должно брать артефакты программного обеспечения из разработки (например, код или документацию) и немедленно поддерживать их, а затем постепенно развивать/поддерживать их в течение жизненного цикла программного обеспечения.

### 1.3 Необходимость технического обслуживания

[ 2 , c1c5]

Техническое обслуживание необходимо для обеспечения того, чтобы программное обеспечение продолжало удовлетворять требованиям пользователей. Сопровождение применимо к программному обеспечению, разработанному с использованием любой модели жизненного цикла программного обеспечения (например, спиральной или линейной). Программные продукты изменяются из-за корректирующих и некорректирующих действий программного обеспечения. Необходимо проводить техническое обслуживание, чтобы

- исправить ошибки;
- улучшить дизайн;
- внедрить усовершенствования;
- интерфейс с другим программным обеспечением;
- адаптировать программы таким образом, чтобы можно было использовать различные аппаратные средства, программное обеспечение, системные функции и телекоммуникационные средства;
- миграция устаревшего программного обеспечения; а также
- удалить программное обеспечение.

Пять ключевых характеристик составляют деятельность сопровождающего:

- поддержание контроля над повседневными функциями программного обеспечения;
- сохранение контроля над модификацией программного обеспечения;
- совершенствование существующих функций;
- выявление угроз безопасности и устранение уязвимостей в системе безопасности; а также
- предотвращение снижения производительности программного обеспечения до неприемлемого уровня.

### 1.4 Большая часть расходов на техническое обслуживание

[ 2 , c4c2 , c5c5 , c7]

На техническое обслуживание уходит основная доля финансовых ресурсов в жизненном цикле программного обеспечения. Распространенное представление о сопровождении программного обеспечения состоит в том, что оно просто исправляет ошибки. Однако исследования и опросы за прошедшие годы показали, что большая часть, более 80 процентов, обслуживания программного обеспечения используется для некорректирующих действий [2\*, рис. 4.1]. Группирование улучшений и исправлений в управленческих отчетах способствует возникновению некоторых неправильных представлений о высокой стоимости исправлений. Понимание категорий обслуживания программного обеспечения помогает понять структуру затрат на обслуживание программного обеспечения. Кроме того, понимание факторов, влияющих на ремонтпригодность программного обеспечения, может помочь сократить расходы. Некоторые факторы окружающей среды и их связь с затратами на обслуживание программного обеспечения включают следующее:

- Операционная среда относится к аппаратному и программному обеспечению.
- Организационная среда относится к политике, конкуренции, процессу, продукту и персоналу.

### 1.5 Эволюция программного обеспечения

Сопровождение программного обеспечения с точки зрения эволюции впервые было рассмотрено в конце 1960-х годов. В течение двадцати лет исследования привели к формулировке восьми «Законов эволюции». Основные результаты включают предположение о том, что техническое обслуживание — это эволюционное развитие, и что решениям по обслуживанию помогает понимание того, что происходит с программным обеспечением с течением времени. Некоторые утверждают, что техническое обслуживание — это непрерывная разработка, за исключением дополнительных входных данных (или ограничений) — другими словами, существующее крупное программное обеспечение никогда не бывает завершенным и продолжает развиваться; по мере своего развития он становится все более сложным, если не предпринимать никаких действий для уменьшения этой сложности.

### 1.6 Категории обслуживания

[ 1 , с3, с6с2] [ 2 , с3с3.1]

Определены три категории (вида) сопровождения: корректирующее, адаптивное и совершенное [2\*, с4с3]. IEEE 14764 включает четвертую категорию — превентивную.

- **Корректирующее обслуживание:** реактивная модификация (или ремонт) программного продукта, выполняемая после поставки для исправления обнаруженных проблем. В эту категорию входит аварийное техническое обслуживание, которое представляет собой внеплановую модификацию, выполняемую для временного поддержания работоспособности программного продукта в ожидании корректирующего обслуживания.
- **Адаптивное обслуживание:** модификация программного продукта, выполняемая после доставки, чтобы сохранить возможность использования программного продукта в измененной или меняющейся среде. Например, может быть обновлена операционная система и могут потребоваться некоторые изменения в программном обеспечении.
- **Полное техническое обслуживание:** модификация программного продукта после поставки для предоставления пользователям дополнительных возможностей, улучшения программной документации и перекодирования для

улучшить производительность программного обеспечения, ремонтпригодность или другие атрибуты программного обеспечения.

- **Профилактическое обслуживание:** модификация программного продукта после поставки для обнаружения и исправления скрытых ошибок в программном продукте до того, как они станут эксплуатационными ошибками.

IEEE 14764 классифицирует адаптивное и совершенное обслуживание как улучшения обслуживания. Он также объединяет категории корректирующего и профилактического обслуживания в категорию корректировки, как показано в таблице 5.1.

Таблица 5.1. Категории сопровождения программного обеспечения		
	Коррекция	Улучшение
Активный	профилактический	совершенный
реактивный	корректирующий	Адаптивный

## 2 ключевые проблемы в обслуживании программного обеспечения

Для обеспечения эффективного сопровождения программного обеспечения необходимо решить ряд ключевых вопросов. Сопровождение программного обеспечения ставит перед инженерами-программистами уникальные технические и управленческие задачи — например, при попытке найти ошибку в программном обеспечении, содержащем большое количество строк кода, разработанного другим инженером-программистом. Точно так же конкуренция с разработчиками программного обеспечения за ресурсы — это постоянная битва. Планирование будущего выпуска, которое часто включает кодирование следующего выпуска и отправку экстренных исправлений для текущего выпуска, также создает проблемы. В следующем разделе представлены некоторые технические и управленческие вопросы, связанные с обслуживанием программного обеспечения. Они были сгруппированы по следующим тематическим рубрикам:

- технические неполадки,
- вопросы управления,
- оценка стоимости и
- измерение.

### 2.1 Технические проблемы

#### 2.1.1 Ограниченное понимание

[ 2 , с6]

Ограниченное понимание относится к тому, насколько быстро инженер-программист может понять, где внести изменения или исправления в программное обеспечение, которое он или она не разрабатывал. Исследования показывают, что около половины всех усилий по сопровождению уходит на понимание программного обеспечения, которое необходимо модифицировать. Таким образом, тема понимания программного обеспечения представляет большой интерес для инженеров-программистов. Понимание сложнее в текстовом представлении — например, в исходном коде, — где часто бывает трудно проследить эволюцию программного обеспечения по его выпускам/версиям, если изменения не документированы и если разработчики не могут их объяснить, что часто бывает. Таким образом, инженеры-программисты могут изначально иметь ограниченное представление о программном обеспечении; многое должно быть сделано, чтобы исправить это.

#### 2.1.2 Тестирование

[ 1 , с6с2.2.2] [ 2 , с9]

Стоимость повторного полного тестирования основной части программного обеспечения значительна с точки зрения времени и денег. Чтобы убедиться, что запрошенные отчеты о проблемах действительны, сопровождающий должен воспроизвести или проверить проблемы, запустив соответствующие тесты. Регрессионное тестирование (выборочное повторное тестирование программного обеспечения или компонента для проверки того, что модификации не вызвали непреднамеренных эффектов) является важной концепцией тестирования при обслуживании. Кроме того, часто трудно найти время для тестирования. Координация тестов, когда разные члены группы обслуживания одновременно работают над разными проблемами, остается сложной задачей. Когда программное обеспечение выполняет критически важные функции, может быть сложно перевести его в автономный режим для тестирования. Тесты не могут выполняться в самом значимом месте — в производственной системе.

#### 2.1.3 Анализ воздействия

[ 1 , с5с2.5] [ 2 , с13с3]

Анализ воздействия описывает, как с минимальными затратами провести полный анализ влияния изменений в существующем программном обеспечении. Сопровождающие должны хорошо разбираться в структуре и содержании программного обеспечения. Они используют эти знания для проведения анализа воздействия, который идентифицирует все системы и программные продукты, затронутые запросом на изменение программного обеспечения, и дает оценку ресурсов, необходимых для внесения изменений. Кроме того, определяется риск внесения изменений. Запрос на изменение, иногда называемый запросом на изменение (MR) и часто называемый отчетом о проблеме (PR), должен быть сначала проанализирован и переведен в программные термины. Анализ воздействия выполняется после поступления запроса на изменение в процесс управления конфигурацией программного обеспечения. IEEE 14764 определяет задачи анализа воздействия:

- анализировать MR/PR; • воспроизвести или проверить проблему;
  - разработать варианты реализации модификации;
  - задокументировать MR/PR, результаты и варианты выполнения;
  - получить одобрение на выбранный вариант модификации.

Серьезность проблемы часто используется для принятия решения о том, как и когда она будет исправлена. Затем инженер-программист определяет затронутые компоненты. Предлагается несколько возможных решений, за которыми следуют рекомендации относительно наилучшего способа действий.

Программное обеспечение, разработанное с учетом удобства обслуживания, значительно облегчает анализ воздействия. Дополнительную информацию можно найти в КА управления конфигурацией программного обеспечения.

#### 2.1.4 Ремонтопригодность

[ 1 , c6s8] [ 2 , c12s5.5]

IEEE 14764 [1\*, c3s4] определяет ремонтопригодность как возможность модификации программного продукта. Модификации могут включать исправления, улучшения или адаптацию программного обеспечения к изменениям в среде, а также изменения в требованиях и функциональных спецификациях.

В качестве основной характеристики качества программного обеспечения ремонтопригодность должна определяться, анализироваться и контролироваться во время разработки программного обеспечения, чтобы снизить затраты на обслуживание. При успешном выполнении ремонтопригодность программного обеспечения улучшится. Часто бывает трудно добиться ремонтопригодности, потому что подхарактеристики часто не являются важным объектом внимания в процессе разработки программного обеспечения. Разработчики, как правило, больше заняты многими другими делами и часто склонны игнорировать требования сопровождающего. Это, в свою очередь, может привести и часто приводит к отсутствию документации по программному обеспечению и средам тестирования, что является основной причиной трудностей в понимании программы и последующем анализе последствий. Наличие системных и зрелых процессов, приемов,

## 2.2 Вопросы управления

### 2.2.1 Согласование с целями организации

[ 2 , c4]

Организационные цели описывают, как продемонстрировать окупаемость инвестиций в деятельность по сопровождению программного обеспечения. Первоначальная разработка программного обеспечения обычно основывается на проектах с определенными временными рамками и бюджетом. Основной акцент делается на том, чтобы предоставить продукт, отвечающий потребностям пользователей, вовремя и в рамках бюджета. Напротив, техническое обслуживание программного обеспечения часто направлено на продление срока службы программного обеспечения как можно дольше. Кроме того, это может быть вызвано необходимостью удовлетворения потребностей пользователей в обновлениях и улучшениях программного обеспечения. В обоих случаях окупаемость инвестиций гораздо менее ясна, поэтому на уровне высшего руководства часто рассматривается основная деятельность, требующая значительных ресурсов и не дающая четкой количественной оценки выгоды для организации.

### 2.2.2 Персонал

[ 1 , c4s5 c10c11]

[ 1 , с4с3, с10с4]

Кадровое обеспечение относится к тому, как привлечь и удержать персонал, обслуживающий программное обеспечение. Техническое обслуживание не часто рассматривается как гламурная работа. В результате персонал, обслуживающий программное обеспечение, часто рассматривается как «граждан второго сорта», и поэтому страдает моральный дух.

### 2.2.3 Процесс

[ 1 , с5] [ 2 , с5]

Процесс жизненного цикла программного обеспечения представляет собой набор действий, методов, практик и преобразований, которые люди используют для разработки и обслуживания программного обеспечения и связанных с ним продуктов. На уровне процесса действия по сопровождению программного обеспечения имеют много общего с разработкой программного обеспечения (например, управление конфигурацией программного обеспечения является важным видом деятельности в обоих случаях). Сопровождение также требует некоторых действий, которых нет в разработке программного обеспечения (подробности см. в разделе 3.2 об уникальных действиях). Эти действия создают проблемы для управления.

### 2.2.4 Организационные аспекты технического обслуживания

[ 1 , с7с2.3] [ 2 , с10]

Организационные аспекты описывают, как определить, какая организация и/или функция будет отвечать за обслуживание программного обеспечения. Команда, разрабатывающая программное обеспечение, не обязательно должна заниматься обслуживанием программного обеспечения после того, как оно будет введено в эксплуатацию.

Принимая решение о том, где будет располагаться функция сопровождения программного обеспечения, организации, занимающиеся разработкой программного обеспечения, могут, например, остаться с первоначальным разработчиком или обратиться к постоянной группе по сопровождению (или специалисту по сопровождению). Наличие постоянной ремонтной бригады имеет много преимуществ:

- допускает специализацию;
- создает каналы связи;
- способствует бескорыстной, коллегиальной атмосфере;
- снижает зависимость от отдельных лиц;
- позволяет проводить периодические аудиторские проверки.

Поскольку у каждого варианта есть много плюсов и минусов, решение следует принимать в каждом конкретном случае. Что важно, так это делегирование или назначение ответственности за техническое обслуживание одной группе или лицу, независимо от структуры организации.

### 2.2.5 Аутсорсинг

[ 3 ]

Аутсорсинг и офшоринг обслуживания программного обеспечения стали крупной отраслью. Организации передают на аутсорсинг целые портфели программного обеспечения, включая обслуживание программного обеспечения. Чаще всего вариант аутсорсинга выбирается для менее важного программного обеспечения, поскольку организации не желают терять контроль над программным обеспечением, используемым в их основной деятельности. Одной из основных задач для аутсорсеров является определение объема требуемых услуг по техническому обслуживанию, условий соглашения об уровне обслуживания и деталей контракта. Аутсорсерам потребуется инвестировать в инфраструктуру обслуживания, а служба поддержки на удаленном объекте должна быть укомплектована носителями языка. Аутсорсинг требует значительных первоначальных инвестиций и настройки процесса обслуживания, который потребует автоматизации.

## 2.3 Оценка стоимости обслуживания

Инженеры-программисты должны понимать различные категории обслуживания программного обеспечения, описанные выше, чтобы решить вопрос оценки стоимости обслуживания программного обеспечения. В целях планирования оценка затрат является важным аспектом планирования

обслуживания программного обеспечения.

### 2.3.1 Оценка стоимости

[ 2 , с7с2.4]

В разделе 2.1.3 описывается, как анализ последствий идентифицирует все системы и программные продукты, затронутые запросом на изменение программного обеспечения, и дает оценку ресурсов, необходимых для выполнения этого изменения.

На оценку затрат на техническое обслуживание влияет множество технических и нетехнических факторов. В IEEE 14764 указано, что «двумя наиболее популярными подходами к оценке ресурсов для обслуживания программного обеспечения являются использование параметрических моделей и использование опыта» [1\*, с7с4.1]. Комбинация этих двух также может быть использована.

### 2.3.2 Параметрические модели

[ 2 , с12с5.6]

Параметрическое моделирование затрат (математические модели) применялось для обслуживания программного обеспечения. Важно то, что для использования и калибровки математических моделей необходимы исторические данные о прошлом обслуживании. Атрибуты драйвера затрат влияют на оценки.

### 2.3.3 Опыт

[ 2 , с12с5.5]

Опыт в форме экспертной оценки часто используется для оценки затрат на техническое обслуживание. Ясно, что лучший подход к оценке технического обслуживания состоит в том, чтобы объединить исторические данные и опыт. Затем рассчитывается стоимость проведения модификации (с точки зрения количества людей и количества времени). Исторические данные оценки технического обслуживания должны предоставляться в результате выполнения программы измерений.

## 2.4 Измерение обслуживания программного обеспечения

[ 1 , с6с5] [ 2 , с12]

Объекты, связанные с обслуживанием программного обеспечения, атрибуты которых могут быть подвергнуты измерению, включают процесс, ресурс и продукт [2\*, с12с3.1].

Существует несколько показателей программного обеспечения, которые могут быть получены из атрибутов программного обеспечения, процесса обслуживания и персонала, включая размер, сложность, качество, понятность, ремонтпригодность и усилия. Меры сложности программного обеспечения также можно получить с помощью доступных коммерческих инструментов. Эти меры представляют собой хорошую отправную точку для программы измерений сопровождающего. Обсуждение процесса разработки программного обеспечения и измерения продукта также представлено в КА процесса разработки программного обеспечения. Тема программы измерения программного обеспечения описана в КА Управления разработки программного обеспечения.

### 2.4.1 Специальные меры

[ 2 , с12]

Сопровождающий должен определить, какие меры подходят для конкретной организации, исходя из собственного контекста этой организации. Модель качества программного обеспечения предлагает меры, характерные для сопровождения программного обеспечения. К мерам подхарактеристик ремонтпригодности относятся следующие [4\*, с. 60]:

- Анализируемость: мера усилий или ресурсов сопровождающего, затраченных на попытку либо диагностировать недостатки или причины отказа, либо определить части, которые необходимо изменить.
- Изменяемость: меры усилий сопровождающего, связанные с реализацией указанной модификации.



- Стабильность: меры неожиданного поведения программного обеспечения, в том числе обнаруженного во время тестирования.
- Тестируемость: мера усилий сопровождающего и пользователей, пытающихся протестировать модифицированное программное обеспечение.
- Другие меры, которые используют сопровождающие, включают
  - размер программного обеспечения,
  - сложность программного обеспечения,
  - понятность и
  - ремонтпригодность.

Предоставление усилий по обслуживанию программного обеспечения по категориям для различных приложений предоставляет бизнес-информацию пользователям и их организациям. Это также может позволить сравнивать профили обслуживания программного обеспечения внутри организации.

### 3 Процесс технического обслуживания

В дополнение к стандартным процессам и действиям по разработке программного обеспечения, описанным в IEEE 14764, существует ряд действий, уникальных для специалистов по сопровождению.

#### 3.1 Процессы технического обслуживания

[ 1 , с5] [ 2 , с5] [ 1 , с5.5]

Процессы обслуживания обеспечивают необходимые действия и подробные входные/выходные данные для этих действий, как описано в IEEE 14764. Действия процесса обслуживания IEEE 14764 показаны на рисунке 5.2. Деятельность по обслуживанию программного обеспечения включает

- реализация процесса,
- анализ проблем и модификаций,
- реализация модификаций,
- обзор/прием технического обслуживания,
- миграция и
- выход на пенсию программного обеспечения.

Другие модели процесса технического обслуживания включают:

- быстрая починка,
- спираль,
- Осборна,
- итеративное улучшение и
- ориентированный на повторное использование.

В последнее время гибкие методологии, которые продвигают легкие процессы, также были адаптированы для обслуживания. Это требование вытекает из постоянно растущего спроса на быстрое выполнение работ по техническому обслуживанию. Усовершенствование процесса обслуживания программного обеспечения поддерживается моделями зрелости специализированных возможностей обслуживания программного обеспечения (см. [6] и [7], которые кратко аннотированы в разделе «Дополнительная литература»).

#### 3.2 Работы по техническому обслуживанию

[ 1 , с5, с6с8.2, с7с3.3]

Процесс сопровождения включает действия и задачи, необходимые для изменения существующего программного продукта с сохранением его целостности. Ответственность за эти действия и задачи возлагается на сопровождающего. Как уже отмечалось, многие действия по обслуживанию аналогичны действиям по разработке программного обеспечения. Специалисты по сопровождению выполняют анализ, проектирование, кодирование, тестирование и документирование. Они должны отслеживать требования в своей деятельности — точно так же, как это делается при разработке — и обновлять документацию по мере изменения базовых планов. IEEE 14764 рекомендует, чтобы при использовании процесса разработки специалист по сопровождению адаптировал его для удовлетворения конкретных потребностей [1\*, с5с3.2.2]. Однако при обслуживании программного обеспечения некоторые действия

включают процессы, уникальные для обслуживания программного обеспечения.

### 3.2.1 Уникальные действия

[ 1 , с3с10, с6с9, с7с2, с7с3 ]

Существует ряд процессов, действий и практик, которые являются уникальными для обслуживания программного обеспечения:

- Понимание программы: действия, необходимые для получения общих знаний о том, что делает программный продукт и как его части работают вместе.
- Переход: контролируемая и скоординированная последовательность действий, в ходе которой программное обеспечение постепенно передается от разработчика к сопровождающему.
- Принятие/отклонение запроса на модификацию: модификации, требующие работы, превышающей определенный размер/усилие/сложность, могут быть отклонены сопровождающими и перенаправлены разработчику.
- Служба поддержки технического обслуживания: координируемая функция поддержки конечных пользователей и технического обслуживания, которая инициирует оценку, приоритизацию и определение стоимости запросов на изменение.
- Анализ воздействия: метод определения областей, затронутых потенциальным изменением;
- Соглашения об уровне обслуживания (SLA), а также лицензии и контракты на обслуживание: договорные соглашения, описывающие услуги и цели в области качества.

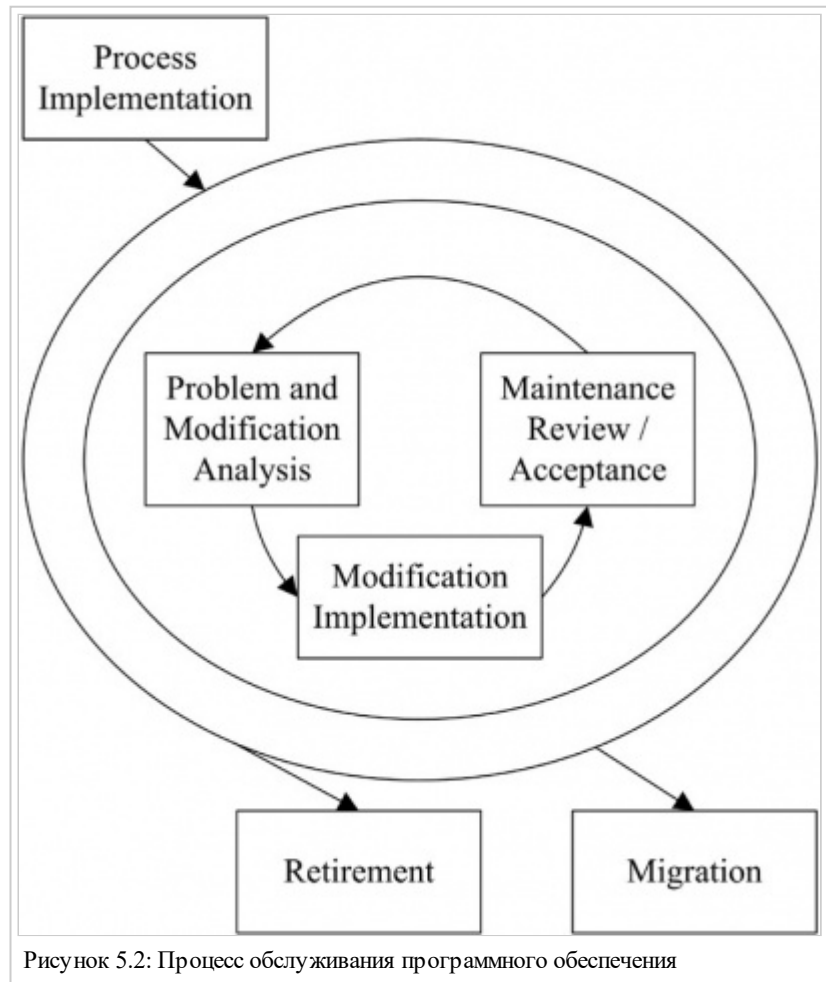


Рисунок 5.2: Процесс обслуживания программного обеспечения

### 3.2.2 Вспомогательная деятельность

[ 1 , с4с1, с5, с6с7 ] [ 2 , с9 ]

Специалисты по сопровождению также могут выполнять действия по поддержке, такие как документация, управление конфигурацией программного обеспечения, проверка и проверка, решение проблем, обеспечение качества программного обеспечения, обзоры и аудиты. Другая важная деятельность по поддержке состоит в обучении сопровождающих и пользователей.

### 3.2.3 Мероприятия по планированию технического обслуживания

[ 1 , с7с3 ]

Важным видом деятельности по сопровождению программного обеспечения является планирование, и специалисты по сопровождению должны решать вопросы, связанные с рядом аспектов планирования, в том числе

- бизнес-планирование (организационный уровень),
- планирование технического обслуживания (переходный уровень),
- планирование выпуска/версии (на уровне программного обеспечения) и

- индивидуальное планирование запросов на изменение программного обеспечения (уровень запроса).

На уровне отдельного запроса планирование осуществляется во время анализа воздействия (см. раздел 2.1.3, Анализ воздействия). Деятельность по планированию выпуска/версии требует, чтобы сопровождающий:

- собирать даты доступности отдельных запросов,
- согласовывать с пользователями содержание последующих выпусков/версий,
- выявлять потенциальные конфликты и разрабатывать альтернативы,
- оценить риск данного выпуска и разработать план отказа на случай возникновения проблем, а также
- информировать все заинтересованные стороны.

В то время как проекты разработки программного обеспечения обычно могут длиться от нескольких месяцев до нескольких лет, фаза обслуживания обычно длится много лет. Оценка ресурсов является ключевым элементом планирования технического обслуживания. Планирование сопровождения программного обеспечения должно начинаться с принятия решения о разработке нового программного продукта и должно учитывать цели в области качества. Следует разработать концептуальный документ, а затем план технического обслуживания. Концепция сопровождения каждого программного продукта должна быть документирована в плане [1\*, с7s2] и должна учитывать

- объем сопровождения программного обеспечения,
- адаптация процесса сопровождения программного обеспечения,
- идентификация организации по сопровождению программного обеспечения, и
- оценка затрат на сопровождение программного обеспечения.

Следующим шагом является разработка соответствующего плана обслуживания программного обеспечения. Этот план должен быть подготовлен во время разработки программного обеспечения и должен указывать, как пользователи будут запрашивать модификации программного обеспечения или сообщать о проблемах. Планирование обслуживания программного обеспечения рассматривается в IEEE 14764. В нем содержатся рекомендации по плану обслуживания. Наконец, на самом высоком уровне обслуживающая организация должна будет заниматься бизнес-планированием (бюджетным, финансовым и кадровым) точно так же, как и все другие подразделения организации. Менеджмент обсуждается в главе «Связанные дисциплины программной инженерии».

### 3.2.4 Управление конфигурацией программного обеспечения

[ 1 , с5s1.2.3] [ 2 , с11]

IEEE 14764 описывает управление конфигурацией программного обеспечения как важнейший элемент процесса обслуживания. Процедуры управления конфигурацией программного обеспечения должны предусматривать проверку, проверку и аудит каждого шага, необходимого для идентификации, авторизации, внедрения и выпуска программного продукта.

Недостаточно просто отслеживать запросы на изменение или отчеты о проблемах. Программный продукт и любые изменения, вносимые в него, должны контролироваться. Этот контроль устанавливается путем внедрения и обеспечения выполнения утвержденного процесса управления конфигурацией программного обеспечения (SCM). КА управления конфигурацией программного обеспечения предоставляет подробные сведения об SCM и обсуждает процесс, с помощью которого запросы на изменение программного обеспечения отправляются, оцениваются и утверждаются. SCM для сопровождения программного обеспечения отличается от SCM для разработки программного обеспечения количеством небольших изменений, которые необходимо контролировать в рабочем программном обеспечении. Процесс SCM реализуется путем разработки и соблюдения плана управления конфигурацией программного обеспечения и операционных процедур. Специалисты по сопровождению участвуют в советах по контролю конфигурации, чтобы определить содержание следующего выпуска/версии.

### 3.2.5 Качество программного обеспечения

[ 1 , с6s5, с6s7, с6s8] [ 2 , с12s5.3]

Недостаточно просто надеяться на повышение качества в результате обслуживания программного обеспечения. У сопровождающих должна быть программа качества программного обеспечения. Это должно быть запланировано, и процессы должны быть реализованы для поддержки процесса обслуживания. Действия и методы для обеспечения качества программного обеспечения (SQA), V&V, обзоров и аудитов должны выбираться вместе со всеми другими процессами для достижения желаемого уровня качества. Также рекомендуется, чтобы специалист по сопровождению адаптировал процессы, методы и результаты разработки программного обеспечения (например, документацию по тестированию) и результаты тестирования. Более подробную информацию можно найти в КА качества программного обеспечения.

## 4 техники ухода

В этом разделе представлены некоторые из общепринятых методов, используемых при обслуживании программного обеспечения.

### 4.1 Понимание программы

[ 2 , с6, с14с5]

Программисты тратят много времени на чтение и понимание программ для внесения изменений. Браузеры кода являются ключевыми инструментами для понимания программы и используются для организации и представления исходного кода. Четкая и краткая документация также может помочь в понимании программы.

### 4.2 Реинжиниринг

[ 2 , с7]

Реинжиниринг определяется как проверка и изменение программного обеспечения для воссоздания его в новой форме и включает в себя последующую реализацию новой формы. Часто это делается не для улучшения ремонтнопригодности, а для замены устаревшего программного обеспечения. Рефакторинг — это метод реинжиниринга, целью которого является реорганизация программы без изменения ее поведения. Он стремится улучшить структуру программы и ее ремонтнопригодность. Методы рефакторинга могут быть использованы во время незначительных изменений.

### 4.3 Обратный инжиниринг

[ 1 , с6с2] [ 2 , с7, с14с5]

Обратный инжиниринг — это процесс анализа программного обеспечения для определения компонентов программного обеспечения и их взаимосвязей, а также для создания представлений программного обеспечения в другой форме или на более высоких уровнях абстракции. Обратный инжиниринг пассивен; это не изменяет программное обеспечение и не приводит к созданию нового программного обеспечения. Усилия по обратному проектированию создают графы вызовов и графы потоков управления из исходного кода. Одним из видов реверс-инжиниринга является повторная документация. Другой тип — восстановление дизайна. Наконец, за последние несколько лет возросло значение обратного проектирования данных, при котором логические схемы восстанавливаются из физических баз данных. Инструменты являются ключевыми для обратного проектирования и связанных с этим задач, таких как повторная документация и восстановление проекта.

### 4.4 Миграция

[ 1 , с5с5]

В течение срока службы программного обеспечения его, возможно, придется модифицировать для работы в различных средах. Чтобы перенести его в новую среду, специалист по сопровождению должен определить действия, необходимые для выполнения миграции, а затем разработать и задокументировать шаги, необходимые для осуществления миграции, в плане миграции, который охватывает требования миграции, инструменты миграции, преобразование продукта. и данные, выполнение, проверка и поддержка. Миграция программного обеспечения также может повлечь за собой ряд дополнительных действий, таких как

- уведомление о намерениях: заявление о том, почему старая среда больше не будет поддерживаться, за которым следует описание новой среды и дата ее доступности;

- параллельные операции: сделать доступными старую и новую среду, чтобы пользователь мог плавно перейти в новую среду;
- уведомление о завершении: когда запланированная миграция завершена, уведомление отправляется всем заинтересованным сторонам;
- обзор после эксплуатации: оценка параллельной работы и влияние перехода на новую среду;
- архивирование данных: хранение старых данных программного обеспечения.

#### 4.5 Выход на пенсию

[ 1 , с5с6]

По истечении срока полезного использования программного обеспечения оно должно быть выведено из эксплуатации. Анализ должен быть выполнен, чтобы помочь в принятии решения о выходе на пенсию. Этот анализ должен быть включен в план выхода на пенсию, который охватывает требования выхода на пенсию, влияние, замену, график и усилия. Доступность архивных копий данных также может быть включена. Прекращение использования программного обеспечения влечет за собой ряд действий, аналогичных миграции.

### 5 инструментов обслуживания программного обеспечения

[ 1 , с6с4] [ 2 , с14]

Этот раздел охватывает инструменты, которые особенно важны при обслуживании программного обеспечения, когда существующее программное обеспечение модифицируется. Примеры, касающиеся понимания программы, включают

- программные слайсеры, которые выбирают только части программы, затронутые изменением;
- статические анализаторы, позволяющие общий просмотр и обобщение содержания программы;
- динамические анализаторы, которые позволяют сопровождающему отслеживать путь выполнения программы;
- анализаторы потоков данных, которые позволяют сопровождающему отслеживать все возможные потоки данных программы;
- перекрестные ссылки, генерирующие индексы компонентов программы; а также
- анализаторы зависимостей, которые помогают сопровождающим анализировать и понимать взаимосвязи между компонентами программы.

Инструменты обратного проектирования помогают процессу, работая в обратном направлении от существующего продукта для создания артефактов, таких как спецификации и описания дизайна, которые затем можно преобразовать для создания нового продукта из старого. Специалисты по сопровождению также используют тестирование программного обеспечения, управление конфигурацией программного обеспечения, документацию по программному обеспечению и инструменты измерения программного обеспечения.

#### ДАЛЬНЕЙШИЕ ЧТЕНИЯ

Эйприл А., Абран А., *Управление обслуживанием программного обеспечения* : оценка и постоянное совершенствование [6].

В этой книге исследуется область небольших процессов сопровождения программного обеспечения (S3M). Он предоставляет дорожные карты для улучшения процессов обслуживания программного обеспечения в организациях. Он описывает конкретную модель зрелости обслуживания программного обеспечения, организованную по уровням, которые позволяют проводить сравнительный анализ и постоянное улучшение. Приведены цели для каждой ключевой области практики, а представленная модель процесса полностью соответствует архитектуре и структуре международных стандартов ISO12207, ISO14764 и ISO15504, а также популярных моделей зрелости, таких как ITIL, CoBIT, CMMI и CM3.

М. Кайко-Маттссон, «На пути к модели обслуживания бизнеса», IEEE Int'l Conf. Сопровождение программного обеспечения [7].

В этом документе представлен обзор Модели зрелости корректирующего обслуживания (CM3). В отличие от других моделей процессов, CM3 является специализированной моделью, полностью предназначенной для корректирующего обслуживания программного обеспечения. Он рассматривает

техническое обслуживание с точки зрения выполняемых действий и их порядка, с точки зрения информации, используемой этими действиями, целей, правил и мотивации их выполнения, а также организационных уровней и ролей, задействованных на различных этапах типичного процесса корректирующего обслуживания.

## ИСПОЛЬЗОВАННАЯ ЛИТЕРАТУРА

- [1] IEEE Std., 14764-2006 (он же ISO/IEC 14764:2006) *Standard for Software Engineering — Software Life Cycle Processes-Maintenance* , IEEE, 2006.
- [2] П. Грабб и А. А. Таканг, *Сопровождение программного обеспечения: концепции и практика* , 2-е изд., World Scientific Publishing, 2003.
- [3] NM Sneed, "Предложение обслуживания программного обеспечения в качестве оффшорной услуги", *Proc. Международная конференция IEEE. Сопровождение программного обеспечения (ICSM 08)* , IEEE, 2008 г., стр. 1–5.
- [4] Дж. В. Мур, *Дорожная карта разработки программного обеспечения: руководство* , основанное на стандартах, издательство Wiley-IEEE Computer Society Press, 2006.
- [5] IEEE, *ISO/IEC/IEEE 24765:2010 Системная и программная инженерия — словарь* , ISO/IEC/IEEE , IEEE, 2010.
- [6] А. Эйприл и А. Абран, *Управление обслуживанием программного обеспечения: оценка и постоянное улучшение* , издательство Wiley-IEEE Computer Society Press, 2008.
- [7] М. Кайко-Маттссон, *На пути к модели обслуживания бизнеса*, *Proc. Международная конф. Сопровождение программного обеспечения* , IEEE, 2001, стр. 500–509.

Получено с " [http://swbokwiki.org/index.php?title=Chapter\\_5:\\_Software\\_Maintenance&oldid=470](http://swbokwiki.org/index.php?title=Chapter_5:_Software_Maintenance&oldid=470) "

- 
- Последнее изменение этой страницы состоялось 24 августа 2015 г., в 19:22.