



ASL 2. Группа процессов сопровождения и обновления приложений.

Часть 1

Мы публикуем выдержки из книги¹, посвященные второй группе процессов – процессам сопровождения и обновления приложений. В нее входят процессы анализа влияния [изменения], проектирования, реализации, тестирования и внедрения. В случае крупных релизов или структурных улучшений приложений объем работ по сопровождению может быть весьма масштабным. Такие обновления происходят в рамках проектного управления.

Тезисы:

- Сопровождение и обновление приложений (далее в тексте иногда употребляется просто термин «сопровождение») – этап, следующий за процессами разработки приложений.
- Степень свободы при сопровождении гораздо меньше, а требования выше. Главная задача заключается в том, чтобы найти идеальное решение в этих непростых условиях. Сопровождаемость приложений чрезвычайно важна в процессах сопровождения и обновления приложений. Нам постоянно приходится сталкиваться с не слишком удачными решениями, но на практике не всегда есть возможности, чтобы улучшить их.
- В мире, где продукты все более основываются на компонентах, значение процессов разработки и сопровождения приложений усиливается. Поэтому согласование с внешней средой становится все более важным аспектом.

¹ Remko van der Pols. ASL 2 - A Framework for Application Management. Van Haren Publishing, 2012. ISBN: 9789087533137. Сейчас готовится к изданию перевод этой книги на русский язык.



Рис. 1.
Процесс сопровождения
и обновления приложений.

Процессы сопровождения и обновления приложений

Структура процессов сопровождения и обновления приложений сопоставима с процессами разработки приложений. Сопровождение и обновление включают в себя следующие процессы:

- **анализ влияния [изменения]** – действия, которые проводятся в рамках ASL, по рассмотрению последствий предложенных изменений и составлению карты этих последствий;
- **проектирование** – анализ информации и проектирование (функциональное);
- **реализация** – осуществление изменений, претворение их в жизнь и компоновка программ (объектов приложений), формирующих приложения;
- **тестирование** – это проверка измененных компонентов (программного обеспечения и услуг) и конечного результата в виде продуктов, которые могут быть приняты и развернуты по желанию заказчика;
- **внедрение** – это подготовка измененного программного обеспечения и других компонентов услуги к использованию. В рамках этого этапа, в частности, уделяется внимание переходу, приемке (тестированию), обучению, инструктажу и переносу программного обеспечения в среду эксплуатации. Он заканчивается формальной приемкой заказчиком (подписание приемочного акта).

Метод и система управления качеством

В значительной степени теоретические и методологические наработки группы процессов сопровождения и обновления приложений были заимствованы из знаний и теорий в области разработки приложений. Эти теории тщательно разрабатывались на протяжении многих лет, появлялись различные новые методы разработки (например, Rational Unified Process или Dynamic Systems Development Method и др.). У них есть ряд сходств и различий в отношении:

- общего подхода (метод водопада, инкрементный метод, прототипирование);
- разделения на этапы (полное, детализированное);
- типов подхода (функциональный, технический);
- схем используемых методов и методологий;
- способов представления данных, функций и временных аспектов (упорядоченности) при проектировании.

Для отдельных проектов или систем, а также на уровне организаций, можно выбрать разные подходы. ASL является процессным фреймворком и поэтому позволяет использовать различные методы. Кроме того, подход редко рассматривается в отрыве от среды, поскольку он также описывает взаимодействие (контрольные точки, входы

и выходы) с внешним миром. Зачастую метод разработки объясняет суть определенных продуктов разработки (например, функциональный дизайн системы, варианты ее использования, документы архитектуры программного обеспечения) и моментов их использования в процессах.

Также методы разработки часто описывают рабочий порядок. В каскадных моделях проектирование представлено в виде четко определенных фаз: сначала выполняется общее планирование, затем следует детальная проработка плана. Вряд ли подобный сценарий возможен в методе разработки динамических систем (DSDM, Dynamic Systems Development Method).

Используемые методы, техники, процессы, этапы процессов и способы интеграции в рабочую среду и организацию – все это составляет неотъемлемую часть системы управления качеством. И выбор этих инструментов может отличаться в зависимости от организации и от используемых приложений. Ответственен за выбор процесс управления качеством.

Различия между разработкой и сопровождением

Как правило, методы и приемы в сопровождении заимствованы из области разработки приложений, а значит, важно еще на начальном этапе разработки принять во внимание требования процессов поддержки, сопровождения и обновления приложений. Иногда этот момент упускают из виду, в результате официальная приемка, которую выполняет организация, управляющая приложением, приводит к дополнительным вложениям и необходимости доработки. Избежать затрат поможет стандартизация выполнения тестирования (см. раздел «Вопросы внедрения» во второй части статьи).

Существуют принципиальные различия между разработкой и сопровождением приложений, хотя при использовании они не бросаются в глаза. Причина в том, что, по сравнению с разработкой, на управление приложениями влияет ряд осложняющих факторов.

- 1. Менее благоприятная отправная точка:** сопровождение затрагивает существующую структуру системы и существующие программы. Изменения в бизнес-процессах или другие технологические возможности могут отодвинуть на второй план доводы в пользу изменения структуры системы. Кроме того, постоянное сопровождение приводит к более сложному программному обеспечению или структурам. Нередко инструменты разработки оказываются не самыми современными и, как правило, не могут обеспечить ту степень поддержки, которая необходима.
- 2. Более высокие требования:** часто новая версия должна быть развернута в среде эксплуатации к определенному моменту времени. Сроки обычно очень сжаты, поскольку либо зависят от изменений законодательства, либо продукты должны выйти на рынок в определенный срок. Кроме того, новая версия должна работать в соответствии с ожиданиями (с точки зрения производительности и надежности), новые функциональные возможности должны быть доступны для использования. Стоит учесть, что при первоначальной разработке можно еще полагаться на предыдущее состояние приложения, к которому можно вернуться.
- 3. Более короткий цикл обратной связи:** проектировщик/разработчик быстрее узнает о своих неудачных решениях. Не самое оптимальное решение останется в системе, пока его не заменят в следующем релизе.
- 4. Меньше возможностей улучшить приложения:** на этом этапе гораздо меньше возможностей улучшить приложения или их управление, нежели во время



Между разработкой и сопровождением приложений есть принципиальные различия, хотя они не бросаются в глаза. Сопровождение похоже на разработку, но на него влияет целый ряд осложняющих факторов

разработки. Причиной этого часто является накопившиеся невыполненные требования, в основном из-за ограниченности финансовых ресурсов, а также из-за неосведомленности менеджмента о том, что подобные улучшения на самом деле выгодны с точки зрения долгосрочной перспективы поддержки системы.

Эти обстоятельства играют большую роль при решении всех вопросов, связанных с сопровождением. По тем же причинам часто приходится обеспечивать сопровождение не в должном объеме, поскольку не всегда удается принимать оптимальные решения. Это означает, что специалисты по управлению приложениями должны творчески подойти к решению задачи и найти компромисс между требованиями, выбранными решениями, уровнем амбиций и осуществимостью изменений, взвесив при этом все предпосылки и риски.

Пересечения процессов поддержки, сопровождения и разработки

За последнее десятилетие значительно уменьшились основные различия между процессами разработки и сопровождения/обновления. Появилось множество гибридных схем, таких как обновление информационных систем и устаревших приложений, разблокировка и интеграция устаревших приложений, инкрементная разработка, определение прототипов, ускоренная разработка приложений (Rapid Application Development) и т. д. В наши дни редко встречается стремление достичь идеальной ситуации за один шаг. Существует определенная степень свободы действий и возможность выбора из множества вариантов, например, с учетом таких параметров, как:

- **цель проектирования.** К чему вы движетесь: к желаемой идеальной архитектуре или к архитектуре, которая была создана на основе существующего информационного обеспечения, а значит, улучшения потребуют меньше шагов?
- **ясность представлений.** Было ли конечное состояние заранее четко обрисовано (спецификация)? Или оно было определено на каком-либо шаге процесса? А может, оно стало очевидно только в конце разработки (прототипирование)?
- **организация работ по сопровождению.** Обновления выполняет команда сопровождения или сотрудники сторонней организации (или и те и другие)?

Сегодня процессы разработки, сопровождения и поддержки пересекаются. Вот какие последствия имеет это пересечение:

- все чаще возникает потребность организовывать эффективную поддержку приложений;



Процессы поддержки и сопровождения и обновления приложений очень тесно связаны друг с другом, поскольку работают с одними и теми же объектами приложений

- возрастает необходимость в знаниях о приложениях и архитектуре используемых приложений, поскольку столь необходимые возможности для развития появляются чаще, чем раньше. В конце концов, сопровождение и обновление используют одни и те же знания – знания существующих приложений;

- логистика программного обеспечения (контроль и распространение программного обеспечения) становится особо важной темой, так как сегодня многочисленные версии могут работать в сочетании друг с другом;
- появляется необходимость в большей эксплуатационной надежности и большем количестве резервных вариантов.

Факторы проектирования и внедрения

Реализация группы процессов сопровождения и обновления приложений зависит от ряда аспектов.

1. Используемый метод разработки (разделение на этапы, контрольные точки).

Метод влияет на то, каким образом выполняется сопровождение или разработка приложений. Это позволяет контролировать этапы, «вехи» (контрольные точки), последовательность операций, время принятия решений и т.д. Выбранный и используемый метод разработки служит отправной точкой для внедрения процессов сопровождения, а также предоставляет инструменты и предварительные условия для организации процессов. Специалисты организации по управлению приложениями не всегда вольны сами решать, какой метод использовать, – он может быть предписан заказчиком или подрядчиком.

2. Ответственность (или ее отсутствие) за интеграцию.

Чаше разрабатывать приложение и сопровождать его приходится в среде, где за техническое обслуживание его компонентов или других приложений, с которыми оно взаимодействует, отвечают сторонние организации по управлению приложениями. Организация, управляющая приложением, может нести или не нести ответственность за интеграцию решений и услуг. Этот факт, а также степень сотрудничества с другими организациями, оказывают влияние на организацию процессов. В частности, от степени взаимодействия с возможными внешними организациями по управлению приложениями зависит реализацию процессов анализа влияния, проектирования, тестирования и внедрения.

3. Управляемость субподрядчиков в направлении достижения желаемой функциональности.

Наличие и управляемость (в плане функциональности) субподрядчиков также влияет на организацию процессов. Если невозможно достичь желаемой управляемости в области функциональности (например, из-за использования типового компонента или платформы), функциональность становится фиксированным фактором при разработке и внедрении приложений. Связанные с этим изменения возникают снова и снова, их добавляют в календарь изменений, чтобы реализовать с помощью процесса управления изменениями. Если же субподрядчики хорошо управляемы, желаемая функциональность (которую нужно изменить) может быть определена в процессе проектирования.

4. Ответственность за определение функциональности и проектирование.

Другой важный фактор – кто отвечает за определение функциональности и насколько просто это происходит. Иногда услуги предоставляются нескольким заказчикам, поэтому пользователи могут объединяться в группы, которые также влияют на определение функциональности.

5. Вид продукта, организации.

Характер приложения и его окружение также играют определенную роль. Организации, которые работают в рамках динамично меняющегося рынка, будут принимать решения быстрее, чем организации, где стабильность, скрупулезность и точность выходят на первый план. В каких-то ситуациях спецификации будут «надежными», в других – станут лишь первой версией в поиске желаемых спецификаций.



Одно из преимуществ ASL в объединении поддержки и сопровождения/обновления приложений на одном уровне управления

Процесс анализа влияния изменения

Воздействие (влияние) предлагаемых изменений регистрируются в процессе анализа влияния. На основании этих данных определяется лучшее направление для реализации изменений.

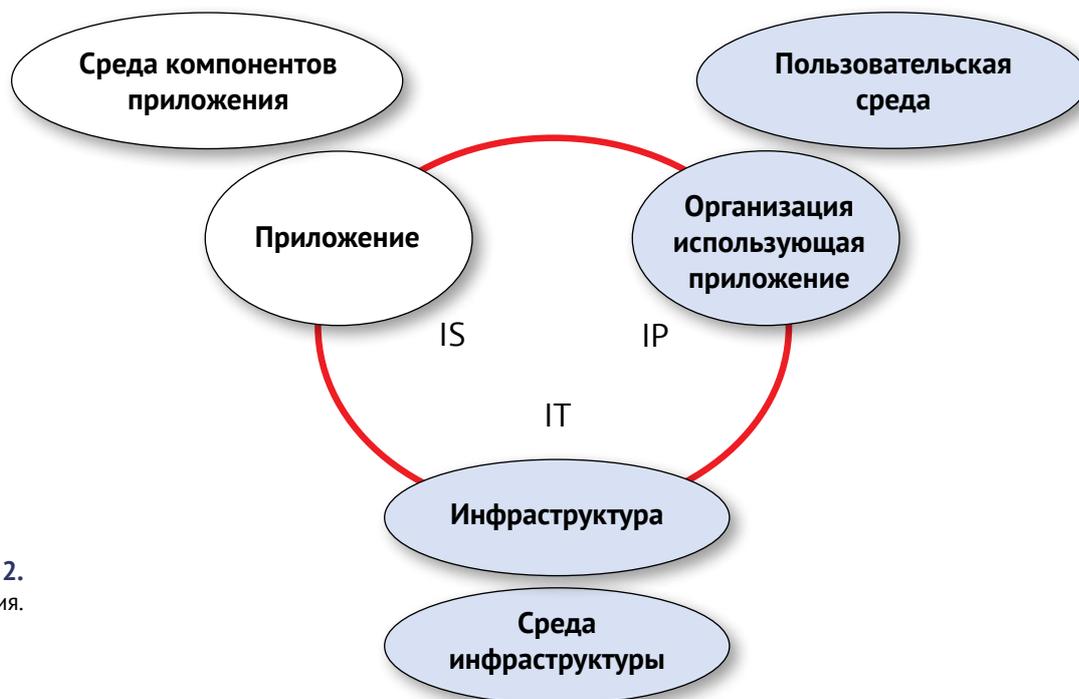


Рис. 2.
Вопросы анализа влияния.

Цель анализа влияния

Цель процесса анализа влияния – эффективно задокументировать достаточно достоверные и точные последствия предложенных изменений с точки зрения трудозатрат, будущих событий, использования и управления, а потом на основе этих данных выбрать лучшее направление реализации изменений.

В процессе управления изменениями собранные изменения группируются в релиз и затем тщательно исследуются в процессе анализа влияния. Анализ влияния определяет воздействие этих изменений на приложения и последующие процессы сопровождения/обновления, а также на среду использования. Воздействие может быть больше или меньше, чем ожидалось. Это означает, что, по результатам анализа влияния, планирование или содержание релиза можно скорректировать в процессе управления изменениями. Другими словами, эти два процесса часто работают циклически.

Вопросы анализа влияния

Процесс анализа влияния фокусируется не только на самом приложении. Большая часть процесса действительно связана с приложением, но учитывая другие аспекты управления информационными системами, внимание уделяется также:

- организации, использующей приложение: какие бизнес-процессы изменятся и что эти изменения повлекут за собой для заказчиков/пользователей?
- инфраструктуре: что значит это изменение для операций, связанных с инфраструктурой и соответствующих соглашений?

Приложения работают в среде, где другие компоненты и информационное обеспечение тоже играют свои роли. К тому же приложение может использовать или состоять из компонентов среды, находящейся за пределами прямого влияния организации, управляющей приложениями. Определение воздействия этой среды и воздействия непосредственно на нее тоже составляет часть анализа влияния.

Приложение и его компоненты. Чтобы оценить это воздействие, в процессе анализа влияния выясняется, какие компоненты (функции, подсистемы, данные) затронут изменения в первую очередь и каковы взаимоотношения между этими объектами. Рассматривается также воздействие на использование компонентов внешних приложений. Очевидна взаимосвязь между процессом анализа влияния и процессом контроля и распространения программного обеспечения. Последний предоставляет средства обслуживания и необходимую информацию.

Одним из результатов анализа влияния является набор изменений, куда входят объекты приложения, которые могут быть преобразованы вследствие изменений. Подробное описание этого можно найти в разделе книги ASL 2 - A Framework for Application Management, посвященном процессу контроля и распространения программного обеспечения.

Инфраструктура и эксплуатация. Чтобы понять, какой эффект оказывают изменения на инфраструктуру, нужно знать, где работает приложение и каковы возможные последствия его изменения. Эта информация может быть получена из процессов поддержки приложений. В данном случае большую роль играет управление конфигурациями, поскольку этот процесс документирует, что и в какой среде используется. Но информация поступает не только из процессов поддержки приложений. В случае масштабных изменений необходимо по возможности проконсультироваться с организацией (организациями), управляющей инфраструктурой.

Организация, управляющая приложениями не всегда в состоянии получить доступ к среде инфраструктуры или к управлению инфраструктурой. Например, если говорить о стандартных приложениях, то в них связь с организациями, управляющими инфраструктурой часто затруднена. То же самое относится и к процессам поддержки приложений в рамках управления приложениями. Этот момент следует выявить и уладить заранее, еще на стадии постановки процесса.

Пользователи. Влияние на пользователей определяется по согласованию с заказчиком¹. То есть необходимо договориться о том, каким образом планируется управлять этим влиянием. В процессе сопровождения в следствие предварительных условий, таких как доступные мощности и сроки, часто делаются определенные уступки, что сказывается на проектировании и полноте решения. Часто это приводит к тому, что различные функциональные возможности остаются неавтоматизированными или частично автоматизированными.

Процесс анализа влияния фокусируется не только на самом приложении, но и на организации, использующей приложения, а также на инфраструктуре, на базе которой он работает



¹Это относится к области управления бизнес-информацией, которая рассматривается отдельно от управления приложениями.

Виды деятельности процесса анализа влияния

Составление общей схемы изменений:

- составление плана изменений, входящих в состав релиза (получение одобрения);
- дальнейшие уточнения для ответа на вопросы, касающиеся изменений.

Оценка изменений:

- определение объектов (конфигурационных единиц), которые будут затронуты предлагаемыми изменениями;
- оценка масштаба изменений в этих объектах в результате предлагаемых изменений;
- выявление взаимосвязей между различными предлагаемыми изменениями и их взаимосвязей с другими релизами (если таковые имеются);
- разработка сценария обработки и тестирования этих изменений.

Оценка последствий:

- оценка последствий для среды эксплуатации, пользовательской среды и для соглашений (требуемая мощность, доступность и т.д.);
- оценка долгосрочных эффектов (таких как сопровождаемость, работоспособность, управляемость, непрерывность и безопасность);
- оценка необходимости дополнительных мер;

- оценка рисков, связанных с внесением изменений;
- оценка масштабов деятельности в результате изменений и оценка сроков.

Верификация и обратная связь:

- верификация для управления бизнес-информацией, процессов поддержки приложений и управления инфраструктурой;
- предоставление информации специалистам, занимающимся процессом управления изменениями.

Управление:

- оценка необходимых ресурсов и метода работы;
- мониторинг развития, соглашений и процессов;
- оценка развития и процесса.

Результаты анализа влияния**Влияние изменений в приложениях (отчет по анализу влияния):**

- предположения и отправные точки для анализа влияния;
- описание изменений и релиза;
- влияние на объекты: затронутые объекты (набор изменений) и влияние на различные характеристики объектов;
- альтернативное решение (или несколько решений);
- действия, которые должны быть выполнены;
- охват/оценка количества трудовых ресурсов, необходимых для релиза;
- возможные риски и меры реагирования на них;
- последствия для пользовательской среды, управления приложениями и среды эксплуатации;
- влияние в долгосрочной перспективе;
- охват изменений (необходимые мощности);
- предлагаемые корректировки к релизу.

Набор изменений:

- затронутые объекты;
- возможные конфликты с другими релизами.

Отчеты о развитии:

- планирование, развитие;
- возможные проблемы, оценки.



*В сущности, все модели неверны,
но некоторые – полезны.
Джордж Бокс*



ASL 2. Группа процессов сопровождения и обновления приложений

Часть 2

Мы продолжаем публиковать выдержки из книги Ремко ван дер Полса «ASL 2 – фреймворк для управления приложениями»¹, посвященные группе процессов сопровождения и обновления приложений. В нее входят процессы анализа влияния изменения, проектирования, реализации, тестирования и внедрения. В первой части статьи мы рассказали об общих принципах и подходах к сопровождению и обновлению приложений, а также подробно описали процесс анализа влияния [изменения]². В этой части мы подробно опишем процессы проектирования, реализации, тестирования и внедрения.

Проектирование

Цель процесса проектирования – формирование и документирование спецификаций информационной системы или ее изменений таким образом, чтобы они могли быть легко реализованы и протестированы. Основным результатом является функциональный или логический проект системы, нетехническое описание ее желаемого функционирования.

Вопросы проектирования

Спецификации. Проектирование начинается с формирования спецификаций, в которых отображено, какой должна быть желаемая функциональность, а также дополнительные параметры безопасности, надежности, производительности, доступности, сопровождаемости и др. (рис. 3). Обычно спецификации получают от заказчиков.

Спецификации не всегда разрабатываются подробно, иногда ситуация не позволяет выявить необходимые детали. Например, если поставщик разрабатывает и производит компоненты, основываясь на ожиданиях, что рынок будет расширяться в определенном направлении. Существует много методов для уточнения спецификаций. Иногда достаточно коротких описаний (простых спецификаций), в иных случаях используется функциональный проект системы, а иногда необходимо использовать технические решения (прототипы). Результатом процесса проектирования является простой и задокументированный проект нового состояния системы или проект, в котором отражены сделанные изменения.

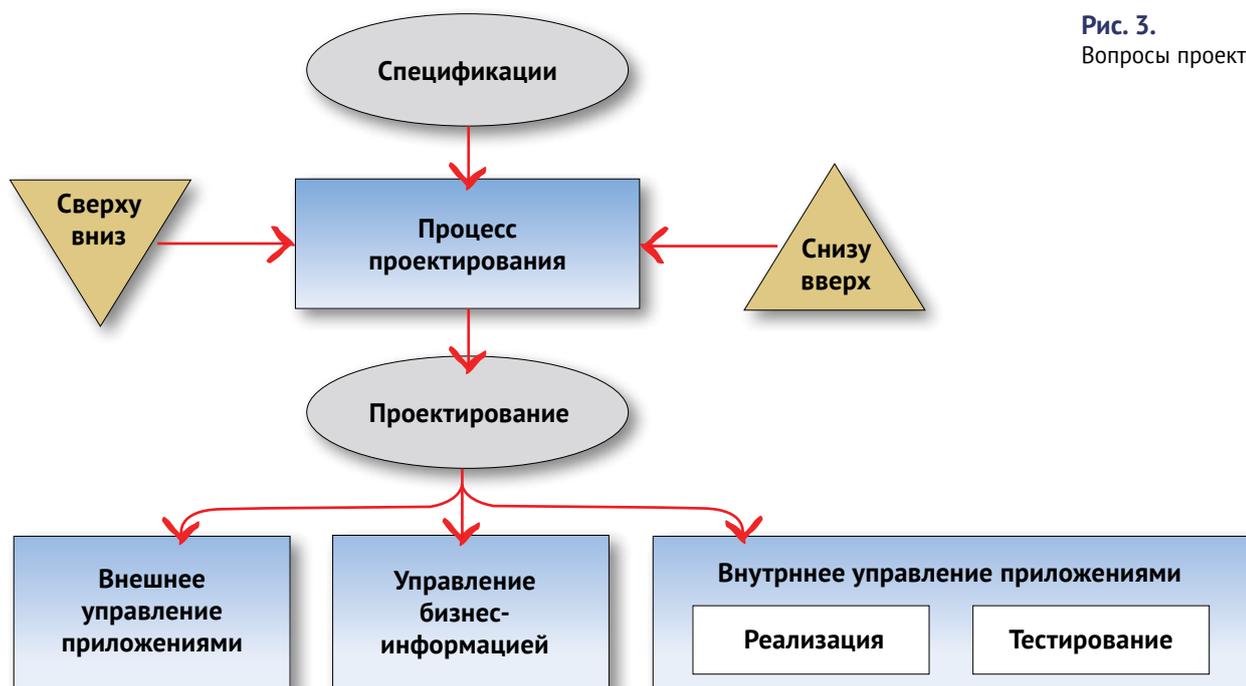


Рис. 3. Вопросы проектирования.

¹Remko van der Pols. ASL 2 - A Framework for Application Management. Van Haren Publishing, 2012. ISBN: 9789087533137. Сейчас готовится к изданию перевод этой книги на русский язык.

² «ASL 2. Группа процессов сопровождения и обновления приложений. Часть 1. Information Management №5 2013.

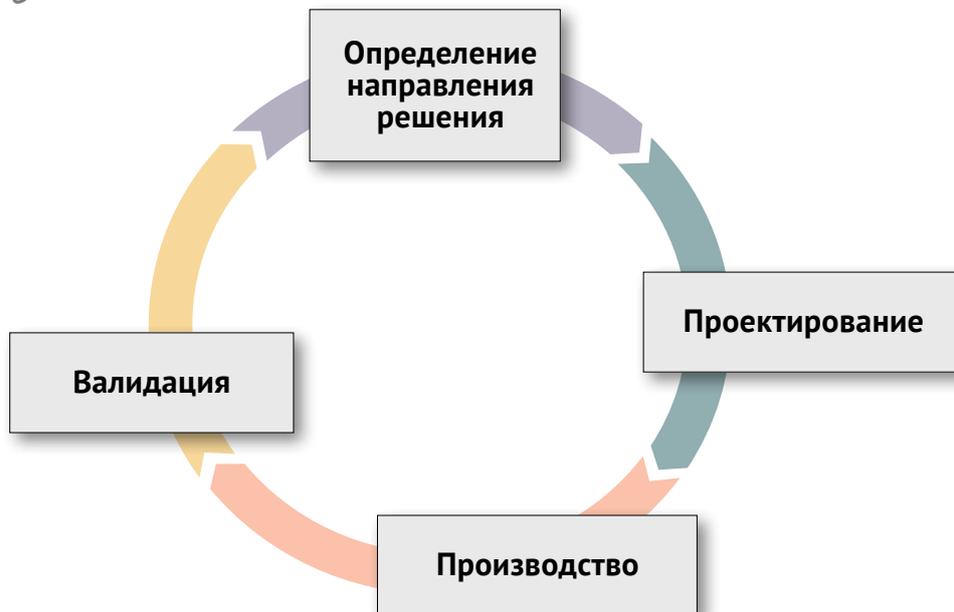


Рис. 4.
Этапы процессов проектирования и реализации.

Процесс проектирования. Процесс проектирования организован аналогично структуре цикла проектирования (рис. 4). Он состоит из четырех этапов:

- **определение направления решения и путей его достижения:** на этом этапе задается желаемое направление функциональных возможностей системы или их изменений;
- **проектирование:** разработка этого пути, составление общего плана решения и проектирование подробной структуры решения;
- **производство:** составление, фиксация и документирование функциональных возможностей, которые будут разработаны или изменены;
- **валидация:** проверка правильности проделанной работы.

В рамках сопровождения приложений преобладает подход «снизу вверх», когда для проведения изменений определяется наименьшая возможная модификация приложения. Если выяснится, что изменение не вписывается в компонент приложения, границы поиска возможностей для модификации расширяются. Таким образом, при подходе «снизу вверх» ищут самый простой и быстрый способ добавить или изменить функциональность в приложении. При разработке новых функциональных возможностей используется, как правило, подход «сверху вниз». Сначала составляют общий план новых возможностей, а затем вносят в него уточнения.

При разработке приложения с нуля проектирование ведется только с использованием подхода «сверху вниз», а при управлении существующими приложениями проектирование изменений обычно проводится по принципу «снизу вверх». Именно поэтому сопровождение приложений должно работать с обоими подходами одновременно.

Формирование и содержание проектной документации. Проектная документация обычно создается с помощью используемых методов разработки/сопровождения. Документация проекта приложения (и изменений) может считаться законченной, если полностью описаны данные, функции (запросы и операции над данными) и временные аспекты (согласованность и последовательность функций). Временные аспекты рассматриваются в сценариях использования приложения. Отметим, что в устаревших системах эти аспекты редко зафиксированы документально. В масштабных приложениях общее проектное решение часто отделено от детального проекта.

Виды деятельности процесса проектирования

Виды деятельности процесса проектирования, как правило, определяются выбранным методом разработки и методологией. Почти во всех методах разработки повторяются следующие шаги:

Разработка запроса:

- углубленный анализ запроса или указанного изменения;

- перевод этой информации в требования и запросы к данным, или в изменения запросов к данным;
- описание соответствующих частей системы.

Определение направления решения:

- определение возможных направлений решения;
- формирование перечня преимуществ и недостатков, их оценка в зависимости от предварительных условий;
- выбор направления решения.

Разработка (детализация) решения:

- определение и перевод спецификаций в проектную документацию;
- формирование спецификаций для функционального тестирования;
- создание документации проекта (в том числе и документации изменений).

Валидация:

- управление внутренним качеством;
- предоставление информации заказчикам;
- согласование с заказчиком.

Управление:

- мониторинг хода работ;
- оценка хода работ, результатов и всего процесса.

Результаты проектирования

Проектная документация:

- системная документация (запись разработанных и согласованных спецификаций в проектную документацию): описание функций (в том числе измененных) приложения, модели данных и процессных потоков;
- возможные изменения в проекте, являющиеся результатом спецификаций;
- спецификации тестирования/план тестирования: описание того, каким образом должны выполняться тесты и какие тестовые сценарии должны использоваться.



Сведения о ходе работ (планирование и контроль и т.д.):

- планирование и ход работ;
- оценки и возможные проблемы.

Процесс проектирования организован аналогично структуре цикла проектирования и состоит из пяти шагов: разработка запроса, определение направления решения, разработка решения, валидация и управление

Реализация

Цель процесса реализации (также называемого «процессом сборки») – преобразование предоставленных проектов или изменений в проектах в конкретные и корректные изменения в автоматизированной информационной системе.

Вопросы реализации

Разделение на этапы. Этапы процесса реализации основаны на той же классификации, которая применяется почти для каждого процесса в группе процессов сопровождения и обновления приложений. Во-первых, проясняются требования (например, путем составления функционального проекта или с помощью прототипа системы). На основании этого в общих чертах определяется, каким образом технически будет выполняться изменение (технический проект системы). Затем прорабатываются детали. После этого изменения реализуются и тестируются.

Чтобы реализовать точные корректные изменения в автоматизированной информационной системе, нужно преобразовать логические спецификации в форму технического решения. Часто это очевидный шаг в тех методах разработки, в которые включена фаза технического проектирования системы.



Проектирование и документирование. Технический проект системы представляет собой описание технических настроек информационной системы или изменений, которые необходимо внести. В техническом проекте системы объясняются принятые технические решения, а также то, как функциональные требования переведены в выбранное техническое решение. Важно заметить, что технический проект системы не всегда можно найти в системной документации. Иногда технические установки и решения интегрированы в функциональный проект системы. В таком случае есть риск, что наряду с управлением бизнес-информацией будут обсуждаться ненадлежащие аспекты (а именно – технические решения).

Изменение осуществляется путем создания или изменения программного обеспечения. Структура ПО и выбранное решение могут оказаться сложными по своему характеру. В таких случаях требуется дополнительная документация. Для этого в исходный код могут быть включены соответствующие комментарии. Другой вариант – доработать или обновить описание программного обеспечения или включить эти комментарии в подробный технический проект системы. Методические указания по использованию этих вариантов предоставляет система управления качеством.

Взаимоотношения с системой контроля программного обеспечения. На первом этапе процесса реализации на основании детальных проектов дается расширенная оценка того, какое программное обеспечение будет затронуто проводимыми изменениями. Результаты этой оценки могут отличаться от ранее выявленного набора необходимых изменений.

Программное обеспечение может быть защищено от изменений: если оно задумано как универсальное, не предполагающее изменений. Также возможно, что характер желаемого изменения окажется таким, что потребуются переработка ядра программного обеспечения или модулей целиком, а не отдельного элемента приложения. Кроме того, не исключено, что придется менять и другое (связанное с изменяемым) программное обеспечение. Подобная ситуация нежелательна, поскольку она повлечет за собой проведение новых проверок, чтобы убедиться, что данные объекты или программное обеспечение не были изменены в то же самое время другим релизом или в результате внепланового технического обслуживания.

Результатом процесса реализации является пакет изменений – объединение программ или элементов данных, которые действительно изменились. После успешного завершения этапа тестирования измененное или новое программное обеспечение следует поместить в среду эксплуатации с помощью процесса контроля и распространения программного обеспечения.

Взаимоотношения с эксплуатацией. В процессе реализации функциональные требования к приложению преобразуются в техническое решение. Эксплуатация решения происходит в рамках управления инфраструктурой. Поэтому, управлению инфраструктурой часто требуется информация, например, о том, какие файлы использует то или иное программное обеспечение, каковы требования для запуска этих систем или программного обеспечения и какая существует зависимость между компонентами в среде промышленной эксплуатации. Эта информация – операционная документация – может быть записана в руководстве по эксплуатации. В любом случае необходимо заключить соглашение с организацией, управляющей инфраструктурой о том, как эта информация будет передана и/или зафиксирована.

Виды деятельности процесса реализации

Определение технического влияния:

- изучение текущей ситуации, желаемых функциональных требований и проекта;
- подробное определение задействованных компонентов.

Проектирование технического решения:

- определение общего направления решения;
- определение группы более детальных изменений;

- определение желаемых изменений;
- обсуждение/проверка изменений;
- разработка спецификаций тестирования;
- документирование технического решения.

Реализация решения:

- изменение программного обеспечения;
- изменение объектов данных;
- изменение возможных дополнительных объектов (например, эксплуатационной документации);
- документирование программного обеспечения.

Тестирование решения:

- тестирование различных приложений или программ для исправления дефектов программного обеспечения (метод «белого ящика»);
- полное тестирование измененного программного обеспечения и файлов в целом.

Управление:

- планирование реализации;
- мониторинг хода работ и реализации, подготовка отчетов о ходе работ;
- оценка и определение проблем или предложений по улучшениям.

Результаты реализации

Объекты, созданные в процессе реализации, состоят из новых или измененных документов, новых или измененных систем, а также результатов тестирования.

Новые или измененные документы:

- технические проекты: описание выбранного технического решения и планов (желательно с дополнительным обоснованием);
- документация к приложению: описание работы программного обеспечения;
- технические спецификации (планы тестирования);
- эксплуатационная документация: информация для управления инфраструктурой (о том, как запустить систему на этапе эксплуатации и соответствующие предварительные условия).

Новые или измененные системы:

- новое или измененное программное обеспечение (предварительные пакеты изменений);
- новые или измененные определения данных (в том числе возможные необходимые преобразования).

Результаты тестов:

- результаты модульного тестирования;
- возможные тестовые данные/скрипты.

Информация управления:

- отчеты о ходе работ;
- оценка процесса и результатов.

Тестирование

Тестирование включает в себя виды деятельности, необходимые для определения, было ли спроектированное действительно реализовано. Тестирование также выявляет, являются ли разработанные или измененные приложения управляемыми и готовы ли они к эксплуатации. **Цель процесса тестирования** – гарантировать, что требуемые изме-



Процесс реализации организован аналогично структуре цикла проектирования и состоит из пяти шагов: определение технического влияния, проектирование технического решения, реализация решения, тестирование и управление

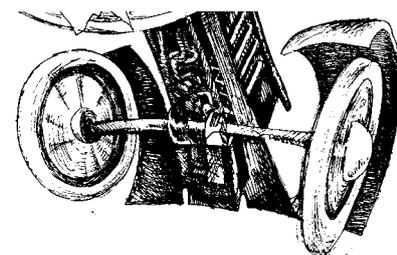


Рис. 5.
Тестирование
в различных средах.



нения реализованы в соответствии со спецификациями, а приложения (после осуществления изменений) функционируют корректно.

Вопросы тестирования

Тестирование является важным шагом на пути предотвращения отказов в работе систем на практике. Для системного решения этой задачи разработаны многие методики и инструменты. ASL не предписывает каких-либо специальных методов или методологий, так как все они, как правило, всегда хорошо адаптируются.

В идеале тестирование следует проводить с первых шагов проведения изменений. В рамках ASL тестирование предусмотрено, в том числе, в видах деятельности процессов проектирования и реализации. Планы тестирования (что именно должно быть протестировано и как тестировать изменения) в числе прочих приводятся в процессе проектирования. С этого начинается процесс, и происходит разработка тестовых сценариев (тестовых данных, тестовых скриптов). Часто имеет смысл сохранить эти тестовые сценарии и адаптировать их или расширять для новых релизов.

Запуск тестов осуществляется в порядке, обратном принятому в процессе внесения изменений. То есть сначала тестируется разработанное программное обеспечение, затем проводится тестирование приложения в целом, и, в конце концов, все компоненты тестируются в своей рабочей среде (рис. 5). Ниже мы вкратце охарактеризуем виды тестирования.

- 1. Модульное тестирование.** В рамках этого тестирования проверяется, соответствуют ли требованиям созданные или измененные элементы программного обеспечения. Этот вид деятельности относится к процессу реализации, поскольку на данном этапе в рамках короткого цикла тестирования определяется, соответствует ли полученный продукт требованиям проекта.
- 2. Интеграционное тестирование или техническое тестирование системы.** Этот вид тестирования проверяет:
 - соответствует ли сформированный модуль разработанному проекту;
 - будет ли изменение действовать в рамках всей системы (то есть тестируются не только измененные части приложения);
 - по-прежнему ли вся система после внесения изменений пригодна к сопровождению и соответствует критериям качества, согласованным со всеми заинтересованными сторонами.
- 3. Функциональное тестирование системы.** Этот вид тестирования выявляет:
 - корректны ли изменения;
 - соответствует ли система в целом согласованной требуемой функциональности;
 - работает ли система как единое целое с функциональной точки зрения;
 - соответствует ли функциональная документация согласованным критериям качества.

4. Производственное (или эксплуатационное) тестирование. В этом виде тестирования определяется (преимущественно эксплуатационной организацией или организацией, управляющей инфраструктурой):

- соответствует ли измененная или обновленная система в эксплуатации (например, в области управления контрактами) желаемым или требуемым основным критериям (например, по времени работы, обработки транзакций и т.д.);
- соответствует ли система дополнительным требованиям (например, необходимой документации по эксплуатации, возможностям настройки и т.д.).

5. Приемочное тестирование. Здесь заказчик или организация, управляющая бизнес-информацией проверяет:

- все ли согласованное реализовано (все ли требования выполнены);
- может ли система быть использована в организации. Таким образом, организация, использующая приложение, тоже проверяет собственную готовность к эффективной работе.

Первый этап тестирования (модульное тестирование) относится к процессу реализации: этот этап представляет собой проверку данного процесса. Заключительный этап – приемочное тестирование – является частью не управления приложениями, а управления бизнес-информацией. Приемочное тестирование проводится в процессе внедрения. Однако обработка его результатов выглядит как обычная процедура в процессе тестирования.

Тестирование в среде, представляющей собой цепочку различных систем и приложений, обычно представляет трудность. Для него не существует отдельного вида тестирования. Если такое тестирование не согласовано и не организовано отдельно, то оно проводится в реальных условиях эксплуатации.

Запуск тестов осуществляется в порядке, обратном принятому в процессе внесения изменений: сначала тестируется разработанное ПО, затем приложение в целом, и, в конце концов, все компоненты тестируются в рабочей среде



Виды деятельности процесса тестирования

Функциональное (логическое) тестирование системы:

- подготовка к тестированию;
- проведение тестирования;
- определение влияния дефектов;
- оценка и определение направления решения;
- исправление дефектов.

Техническое тестирование системы:

- подготовка к тестированию (создание вариантов тестирования или регулировка наборов тестов);
- проведение тестирования;
- определение влияния дефектов;
- оценка и определение направления решения;
- исправление дефектов.

Поддержка производственного (эксплуатационного) тестирования:

- предоставление поддержки при проведении тестирования;
- определение влияния дефектов;
- оценка и определение направления решения;
- исправление дефектов.

Управление:

- составление отчета о развитии;
- составление отчета об ошибках;
- оценка процесса и результатов.



Результаты тестирования

Продукты тестирования:

- результаты тестирования (ожидаемые результаты, фактические результаты);
- протоколы тестирования (количество выявленных ошибок, статусы, срок завершения);
- ошибки и запросы.

Управленческие отчеты:

- отчеты о ходе работ;
- оценка.

Поддержка тестирования:

- варианты тестирования/наборы тестов;
- программное обеспечение для проведения тестирования/тестовые скрипты.

Внедрение

Процесс внедрения охватывает все виды деятельности, которые должны быть проведены для того, чтобы внести изменения, предложенные процессом управления изменениями, в реальное использование и обработку данных. **Цель процесса внедрения** – выполнить необходимые предварительные условия, чтобы в дальнейшем безошибочно применять новую версию приложения и завершить процесс сопровождения.

Вопросы внедрения

Внедрение приложения в принципе не должно требовать большого количества усилий от управления приложениями, поскольку оно уже было разработано, протестировано и признано правильным. Скорее усилия потребуются со стороны внешних участников: пользователи должны быть подготовлены, программное обеспечение должно быть установлено в инфраструктуру, и, возможно, инфраструктура должна быть адаптирована соответствующим образом (рис. 6).

Поддержка со стороны управления приложениями заключается в следующем:

- поддержка при развертывании для использования организацией;
- поддержка при вводе системы в эксплуатацию силами управления инфраструктурой;
- поддержка использования или интеграции измененной функциональности и программного обеспечения другими организациями, управляющими приложениями;
- завершение изменений и архивирование документации приложения и проектной документации, для использования в собственных процессах.

Эти четыре вида деятельности актуальны для внутренней организации. Для команды, осуществляющей управление приложениями, внедрение является первым практическим шагом к управлению программным обеспечением.

Пример различных релизов

Релиз 12–01 является комплексным. Модель данных в этом релизе претерпела некоторые серьезные изменения. По этой причине были запущены широкомасштабные программы для преобразования данных в новую структуру. Используемое для этого сложное программное обеспечение потребовало значительной поддержки со стороны организации, управляющей приложениями. Релиз 12–01 также повлек за собой проверку качества данных на значительно более высоком уровне.

Это оказало большое влияние на администраторов бизнес-информации и пользователей. Поскольку данные должны быть проверены и исправлены, в ходе подготовки к внедрению организация, управляющая приложениями, разработала и использовала программное обеспечение для мониторинга, чтобы выявить возможные ошибки. Для этого пользовательская организация сделала дополнительные запросы. Выполнение этих работ также существенно повлияло на среду эксплуатации, и потребовало серьезной поддержки.

Релиз 12–02 стал простым релизом в рамках сопровождения без каких-либо преобразований и переносов данных.



Рис. 6.
Вопросы внедрения.

Другим важным этапом является приемочное тестирование. С его помощью менеджер по бизнес-информации сможет проанализировать, все ли изменения были выстроены правильно с точки зрения пользователя. Приемочное тестирование является основанием для составления акта приемки работ по договору. Также приемочное тестирование подтверждает, соответствует ли приложение требованиям, установленным организацией, управляющей приложениями.

Внедрение приложения в принципе не должно требовать большого количества усилий от управления приложениями, поскольку оно уже было разработано, протестировано и признано правильным. Скорее усилия потребуются со стороны внешних участников

Деятельность, которая будет осуществляться в рамках процесса внедрения, зависит от ряда факторов:

- набора согласованных услуг;
- характера приложения и его возможностей;
- характера изменений, способа их осуществления и их влияния (одни изменения требуют сложного переноса данных, другие – нет).

Виды деятельности процесса внедрения

Поддержка перехода в эксплуатационную (производственную) среду:

- поддержка подготовки процесса установки и эксплуатации;
- поддержка (или осуществление) изменения данных в производственной среде, поддержка технических преобразований;
- подготовка и поддержка разработки графика обработки данных;
- подготовка фактической передачи в среду эксплуатации (производственную).

Поддержка организации, использующей приложение:

- поддержка при подготовке приемо-сдаточного тестирования;
- поддержка во время проведения приемо-сдаточного тестирования (данные, вопросы, обработки результатов тестирования);
- поддержка управления бизнес-информацией, при внедрении в организацию: под-



Эксплуатация и критерии приемки

При внедрении важно проверить, могут ли быть развернуты новые приложения или новый релиз. В ходе тестирования управляемости и обслуживаемости приложений важно заранее выявить препятствия, которые помешают проводить процессы поддержки, сопровождения и обновления приложений, и вовремя исправить их. Тестирование поддержки, сопровождения и обновления, как правило, проводится в рамках процесса тестирования.

Если разработка осуществляется внешними организациями (не той, которая осуществляет поддержку, сопровождение и обновление), понадобятся дополнительные шаги и тестирование, потому что в данном случае происходит передача ответственности за объекты. В этом случае необходимо провести отдельное тестирование, чтобы убедиться, что другие стороны, участвующие в разработке, тоже создали приложения и/или программное обеспечение, которое отвечает критериям поддержки, сопровождения и обновления. Выполнение тестирования может быть передано принимающей стороне. Как правило, в этом случае организация потребует гарантий того, что заявленные критерии были и раньше учтены в этом процессе.

поддержка изменений в руководствах пользователя, поддержка во время функциональных преобразований (параметры функционального контроля).

Подготовка к завершению релиза:

- архивирование документов, подготовка к выпуску протокола, инициирование процедуры оценки;
- исправление ошибок, найденных во время приемочного тестирования.

Завершение задания:

- подтверждение выполнения обязательств;
- предоставление производственного заказа;
- обновление статуса в управлении изменениями.

Управление:

- оценка необходимых способностей и методов работы;
- объединение отчетов о ходе работ;
- проведение оценки.

Результаты внедрения

Поддержка развертывания:

- поддержка приемочного тестирования;
- возможные обращения по результатам приемочного тестирования;
- поддержка введения в организацию, использующую приложение.

Поддержка развертывания системы в среде эксплуатации:

- обеспечение выполнения требований поддержки, сопровождения и обновления приложений и эксплуатационной среды;
- дополнительная информация о среде эксплуатации;
- возможные изменения в планировании производственной деятельности;
- поддержка изменения данных и преобразований.

