

МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ
ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ БИЗНЕС СИСТЕМ
КАФЕДРА Системный анализ и управление в информационных
технологиях

На правах рукописи

Щербатых Ирина Викторовна

УПРАВЛЕНИЕ ТРЕБОВАНИЯМИ К АВТОМАТИЗИРОВАННОЙ
СИСТЕМЕ ФОРМИРОВАНИЯ КОНСОЛИДИРОВАННОЙ
ОТЧЁТНОСТИ

Выпускная квалификационная работа

Студент: Щербатых И.В.

**Научный руководитель: д.ф.-м. н., профессор кафедры вычислительной
математики механико-математического факультета**

МГУ им. М.В. Ломоносова, Кумсков М.И.

Консультант:

к.э.н., начальник отдела бюджетирования, Алмазов С.Е.

Рецензент: **степень, звание, ФИО**

Зав. кафедрой: д.т.н., профессор, Калянов Г.Н.

Москва 2010 гг.

АННОТАЦИЯ

__ чертежей, 136 с., __ табл., __ рис., 5 прил., 29 источников

Ключевые слова:

Управление требованиями, трассируемость, изменения в требованиях, управление изменениями, репозиторий требований.

Объект исследования – задача управления требованиями в проекте создания автоматизированной системы консолидированной отчётности.

Цель работы – предложить решение задачи автоматизации управления требованиями в реальном проекте создания программного обеспечения.

Методы исследования: причинно - следственный анализ.

Результат – описан процесс по автоматизации работы с требованиями в проекте АСКО, разработан регламент работы с требованиями, создана база данных с требованиями проекта АСКО, шаблонами документов с требованиями, созданы варианты использования системы управления требованиями.

Новизна – предложен и описан процесс автоматизированной поддержки требований на этапе сопровождения системы АСКО в терминах ITIL/ITSM

Апробация: научная работа опубликована в сборнике 2 Научно-практической Конференции Академии ИБС, возможно внедрение решения на этапе сопровождения системы АСКО.

Эффект от внедрения решения: экономия, полученной в результате внедрения решения может быть косвенно оценена как экономия ресурсов, которые могли бы потребоваться организации для выполнения задач, возникающих в альтернативном варианте.

СОДЕРЖАНИЕ

1	Введение	5
2	Глава 1 “Проблемы в управлении требованиями”	8
	2.1 Введение “Разработка программного обеспечения (ПО) в Век Процессов”	8
	2.2 Зачем управлять требованиями	9
	2.3 Что такое Требование?	10
	2.4 Что такое Управление Требованиями?	11
	2.5 Проблемы в Управлении Требованиями.	12
	2.6 Методики УТ и их применение на практике	14
	2.7 Важные понятия о требованиях	19
	2.8 Управление требованиями на практике.	25
	2.9 Управление требованиями: потоки работ.	27
	2.10 Средства УТ. Лидеры и остальные.	41
	2.11 Выводы	45
3	Глава 2 “Управление требованиями в АСКО”	47
	3.1 Постановка задачи на проектирование СУТ	48
	3.2 Определение процесса управления требованиями	55
	3.3 Шаблоны документов	67
	3.4 Инструментальная поддержка процесса управления требованиями. .	71
	3.5 Выводы	72
4	Глава 3 “Управление требованиями на этапе сопровождения АСКО”. .	74
	4.1 Процесс управления изменениями в терминах ITIL	74
	4.2 Основные термины	75
	4.3 Цель процесса.	78
	4.4 Процесс.	79
	4.5 Виды деятельности в рамках Процесса Управления Изменениями. .	85
	4.6 Контроль процесса	99
	4.7 Выводы	100

5	Глава 4 Оценка эффекта от внедрения СУТ на базе Rational RequisitePro.....	103
5.1	Методика исследования прецедентов окупаемости инвестиций.....	103
5.2	Выводы.....	121
6	Заключение.....	126
7	Список использованной литературы.....	128
8	Перечень сокращений.....	131
9	Приложение 1 “Постановка задачи на проектирование СУТ”.....	133
10	Приложение 2 “Причинно-следственный анализ”.....	133
11	Приложение 3 “Процесс управления требованиями в проекте АСКО”.	135
12	Приложение 4 “План управления требованиями”.....	136
13	Приложение 4 “Сценарии использования СУТ”.....	136
14	Приложение 5 “Репозиторий АСКО. Финальная версия”.....	136

1 Введение.

В процессах разработки ПО термин "управление требованиями" до сих пор представляется противоречивым сочетанием слов. В то же время, отсутствие четкого и постоянно обновляемого определения требований является одной из основных причин провала проектов. Проблемы решаются проще всего, если они обнаруживаются на ранних стадиях жизненного цикла разработки, – особенно, если исправления требуют изменений архитектуры ПО. Хотя большинство ИТ-специалистов понимают значение выработки четких и лаконичных требований и их ведения на протяжении всего жизненного цикла проекта, достигнуть этого в действительности достаточно сложно.

Сбор всех требований, которые предъявляются к разрабатываемым программным системам (ПС), и управление ими – это весьма сложная задача.

Среди главных причин этой сложности можно выделить следующие:

- большое число потенциальных заинтересованных лиц, характерное для проектов разработки ПС, требования которых нужно выявить и зафиксировать;
- разнообразие типов требований, каждый из которых требует специфического описания, своих атрибутов и степени детализации. Например, пожелания заинтересованных лиц, функциональные и нефункциональные требования к системе и т.п.;
- необходимость создания и поддержания сложной иерархической структуры;
- необходимость трассировать требования, то есть выявлять и фиксировать взаимосвязь между требованиями различных типов. Например, между пожеланиями заинтересованных лиц, функциональными требованиями в виде сценариев использования и тестовыми требованиями, которые будут использоваться при передаче системы заказчику;

- требования меняются в ходе выполнения проекта. Причин для этого много и большинство из них не сводятся к чьим-то ошибкам.

Цель данной работы заключается в решении задачи по автоматизации процесса управления требованиями (УТ) в проекте создания автоматизированной системы по формированию консолидированной отчётности ОАО “Газпром” (АСКО).

Задачи, решаемые в работе по управлению требованиями, включают в себя методические (разработка план управления требованиями для системы АСКО, описывающего процесс УТ, проанализирована структура Проектного Офиса и выделены документы Проекта АСКО, шаблоны которых будут использоваться при управлении требованиями.) Представлены EPC и UML – диаграммы, наглядно демонстрирующие бизнес-процесс УТ, как есть (“AS IS”) и как должен стать после автоматизации (“TO BE”). Практические задачи включают создание репозитория в RequisitePro для проекта АСКО, Проекта АСКО, введение и размещение документов с требованиями для АСКО, введение глоссария проекта, бизнес-требований к системе, функциональных требований, создание сценариев использования инструмента УТ, предложение бизнес-процесса и его инструментальной поддержки для управления изменениями на этапе сопровождения АСКО.

Научная новизна данной работы состоит в совершенствовании существующих методов проектного управления, в частности, управления требованиями, для задач анализа, оптимизации работ с требованиями, обработки информации в результате изменений требований, с целью повышения надёжности и эффективности функционирования автоматизированной системы по формированию консолидированной отчётности, создаваемой подразделением IBS.

Апробация результатов исследования возможна при внедрении решения на этапе сопровождения проекта АСКО.

Основные положения работы опубликованы в сборнике Второй ежегодной научно-практической конференции “Информационные бизнес-системы”.

Первая глава работы посвящается обзору проблематики процесса управления требованиями с описанием известных научно-практических работ в этой области, даны необходимые понятия и определения. Дано развёрнутое описание предметной области с обозначением проблемных мест, приведено функционально-стоимостное сопоставление соответствующих информационных систем, способствующих решению задачи управления требованиями.

Во второй главе сформирована постановка актуальной прикладной задачи подразделения IBS, в котором выполнялась работа, и описаны методы ее решения, используемые инструментальные средства. Приведено решение задачи, выполненное лично автором, представленное в виде методики управления требованиями в проекте, use-case моделей, описывающих работу с репозиторием и сам репозиторий с требованиями проекта, шаблонами документов проекта, которые позволят получить новые результаты в области проектного управления.

В третьей главе даются рекомендации по тиражированию результатов решения задачи на этапе сопровождения системы, описан процесс управления изменениями требований в терминах ITIL/ITSM.

В четвертой главе приводится методика оценки эффекта от внедрения решений на базе продукта IBM Rational RequisitePro, предложенная компанией IDC.

2 Глава 1 “Проблемы в управлении требованиями”.

2.1 Введение “Разработка программного обеспечения (ПО) в Век Процессов”.

Большинство команд - разработчиков ПО в 1990-е годы интенсивно применяли процессный подход к разработке ПО. В это время стали вводиться и широко распространяться стандарты оценки и сертификации эффективных процессов разработки программного обеспечения, было опубликовано множество книг и статей по процессам разработки ПО, соответствующие материалы по моделированию и реинжинирингу бизнес-процессов. Огромное количество программных инструментов помогало выбирать и применять на практике эффективные процессы разработки. Глобальная зависимость экономики от программного обеспечения на протяжении десятилетия ускоряла и благоприятствовала развитию процессов и улучшению систем качества.

Как же объяснить высокий уровень провалов проектов разработки ПО в нашем веке? Почему много, а точнее большинство, проектов всё ещё страдают от увеличения сроков, перерасхода бюджета и проблем с качеством? Как можно улучшить качество разрабатываемых систем, от которых всё больше и больше зависят бизнес, национальная экономика и повседневная жизнь?

Ответом на вопросы, как всегда, становятся люди, инструменты и те процессы, что применяются в отрасли. Управление требованиями часто предлагается в качестве решения текущих проблем разработки программного обеспечения, пока ещё относительно мало внимания уделялось совершенствованию этой дисциплины на практике.

В данной главе представлены элементы эффективного процесса управления требованиями и освещаются некоторые трудности для успешного осуществления.

Управление требованиями в равной степени относится к чисто программным проектам и к проектам, в которых ПО является лишь частью результата или не включается в результат проекта вообще. Для удобства, в этой главе в дальнейшем будет использован термин “система”, обозначающий такого рода проекты (один или все вместе). Также внимание уделялось абстрактному характеру разработки ПО как таковому, так и в сочетании с разработкой архитектуры (что усложняет управление требованиями).

2.2 Зачем управлять требованиями

Самый простой ответ: команды-разработчики, которые могут управлять и управляют требованиями, делают это ради успеха проекта. Удовлетворение требований их проекта определяет успех. Неспособность управлять требованиями уменьшает вероятность достижения этих целей.

В поддержку последних утверждений:

- В отчёте The Standish Group’s “CHAOS Summary 2009” установлено, что значительный вклад в “провал” проекта внесло неумение управлять требованиями. [1]
- В декабре 1997 Computer Industry Daily опубликовало результаты исследования Sequent Computer Systems, Inc. при участии 500 ИТ – менеджеров из США и ОК, которое показало, что 76 процентов опрошенных испытывали полный провал проекта в течение своей карьеры. Наиболее часто называют причиной провала проекта “изменение потребностей пользователя”. [2]

Достаточной мотивацией для управления требованиями должно стать желание избежать неудачи, провала проекта. Также могут мотивировать увеличение вероятности успешной реализации проекта и другие преимущества от управления требованиями. В отчёте The Standish Group также установлено, что управление требованиями и есть фактор обеспечивающий успешность проекта.

Статистика успешности выполнения программных проектов по данным компании Standish Group в США за 2000 год: только 28% проектов были успешными, 49% - завершились с превышением лимита бюджета или времени, а 23% потерпели полный крах и были прекращены. Среди факторов провала: отсутствие вовлечения пользователей в проект, нечеткие цели, неполные требования и спецификации, постоянное изменение требований и спецификаций, ошибки планирования.

2.3 Что такое Требование?

Требование – это любое условие, которому должна соответствовать разрабатываемая система или программное средство (ПС). Таким требованием может быть и возможность, которой система должна обладать, и ограничения на ее производительность или пользовательский интерфейс, и нормативные требования, которым она должна удовлетворять. Управление требованиями – необходимое условие успешной разработки ПС.

Первым шагом к пониманию управления требованиями является соглашение о ведении общего словаря – глоссария. Определение требования в RUP: “условие или возможность, которому система (разрабатываемая) должна удовлетворять”. В Институте Электроники и инженеров-электриков (IEEE) используется подобное определение.

Известные авторы инженерии требований Мерлин Дорфман и Ричард Г. Тайер предлагают совместимое и более уточнённое определение, специфичное (но не обязательно ограничиваемое) для области программного обеспечения.

“Требование к программному обеспечению может быть определено как:

- Возможность программного обеспечения, необходимая пользователю для решения проблемы или достижения поставленной цели.
- Возможность программного обеспечения, которая должна выполняться или которой должна обладать система или компонент системы, чтобы

удовлетворять контракту, спецификациям, стандартам или же официально утвержденным документам”. [3]

2.4 Что такое Управление Требованиями?

Поскольку требования – это то, чему разрабатываемая система должна соответствовать, и соответствие некоторому набору требований может определить провал или успех проекта, имеет смысл выявить те требования, которые необходимо записать, организовать и отслеживать их в случае их изменения.

С другой стороны, управление требованиями – это:

- Систематический поход к выявлению, организации и документированию требований к системе, и
- Процесс, который устанавливает и поддерживает соглашение об изменениях требований к системе между заказчиком и проектной группой.

Это определение подобно определениям Дорфмана – Тайера и IEEE “инженерии программных требований”. Инженерия требований включает в себя выявление, анализ, спецификацию, проверки и управление требованиями к ПО, под управлением требованиями понимается планирование и контроль над вышеупомянутыми мероприятиями. [4] Все эти мероприятия входят в состав определения управления требованиями, принятого в Rational Software. Основное отличие определений состоит в употреблении слова “управление”, предпочитаемого “инженерии”. Управление - более подходящее определение для всех входящих в его состав мероприятий, оно точно подчёркивает важность отслеживания изменений для сопровождения соглашения между заинтересованными лицами и проектной командой.

Термин “извлечение” определяется как комплекс мероприятий, в течение которых команды работают над выявлением или обнаружением запросов заинтересованных лиц, определением реальных потребностей из

запросов и достижением удобного набора требований, которые могут быть размещены в системе для удовлетворения определенных потребностей.

2.5 Проблемы в Управлении Требованиями.

Какие могут возникать сложности с процессом, который должен обеспечить соответствие разрабатываемой системы ожиданиям заказчиков? На практике в реальных проектах возникают некоторые сложности. **Error! Reference source not found.** отражает результаты исследования 1996 года разработчиков, управленцев и менеджеров по качеству. Представленная статистика показывает процентное соотношение самых распространенных проблем с требованиями у респондентов.

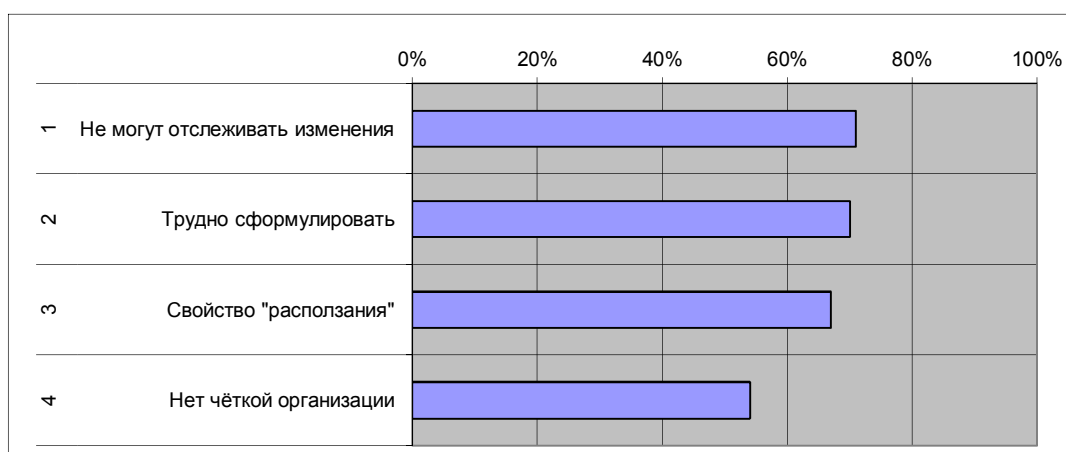


Рисунок 2-1 Основные проблемы с требованиями

Полный список проблем включает в себя:

- Требования не всегда очевидны и имеют много источников.
- Требования не всегда можно четко сформулировать словесно.
- Существует множество типов требований на разных уровнях детализации.
- Число требований может стать неподдающимся обработке, если его не контролировать.
- Требования связаны одно с другим, с различными результатами процессов различными способами.

- Требования имеют уникальные свойства или характеристики, например, они не являются одинаково важными, не все в равной степени легко удовлетворить.

- Из-за большого числа заинтересованных и ответственных сторон, требованиями должны управлять межфункциональные группы людей.

- Требования меняются.

- Требования могут быть срочными.

Когда эти проблемы объединяются с неадекватным управлением требованиями и производственными навыками, с отсутствием простых в использовании инструментов, многие команды теряют надежду на хорошее управление требованиями. Rational Software разработала методологию по обучению команд процессам управления требованиями. Кроме того, Rational Requisite Pro® является доступным инструментом автоматизации эффективного управления требованиями.

IBM Rational Unified Process позволяет компании-разработчику настраивать весь процесс разработки ПО. В отличие от большинства современных методологий или требований к процессу разработки, ориентированных на строго определенный уровень формализации процесса (как правило, либо очень высокий, либо напротив, очень низкий), RUP позволяет получить именно тот уровень формализации, который необходим в проекте.

При разработке ПО для государственных предприятий и в ряде других ситуаций компании-разработчику приходится выполнять формальные требования отечественных и международных стандартов (ГОСТы серий 19 и 34, ГОСТ ИСО/МЭК 12207 и др.). В этом случае можно настроить RUP на достаточно высокий уровень формализации процесса. Более того, при использовании инструментальных средств IBM Rational несложно разработать шаблоны документов, которые будут создаваться автоматически с помощью IBM Rational SoDA на основе стандартных моделей и артефактов и соответствовать требованиям ГОСТ.

2.6 Методики УТ и их применение на практике

Неконтролируемые изменения требований в ходе жизненного цикла разработки неизменно приводят к путанице, срыву графика работ и перерасходу выделенного бюджета. Неудовлетворенными оказываются все – конечные пользователи, аналитики, разработчики и менеджеры. Такие жалобы, как "Пользователи сами не знают, чего хотят", "Приложение не удовлетворяет нашим потребностям" и "Этот проект требует слишком много времени и средств", являются очень распространенными.

Управление требованиями (УП) часто упускается из виду или не применяется должным образом в проектах по разным причинам.

Некоторые компании считают, что документы Word, электронные письма, телефонные звонки и совещания с участием заинтересованных лиц сами по себе достаточны для управления требованиями. Фактически, это и есть та методика, которая в настоящее время используется во многих ИТ-компаниях для большинства проектов. Однако, неформальный, эпизодический характер этого подхода не гарантирует строгого определения требований, которое доводится до сведения всех заинтересованных лиц и понимается ими. Кроме того, этот метод не позволяет управлять неизбежными изменениями требований, которые возникают в ходе проекта. Цель заключается не в предотвращении изменения, а в его принятии и управлении им. В современном мире сложных многоуровневых систем, разнотипных и рассредоточенных групп разработчиков, приложений, имеющих критический характер для бизнеса и деятельности предприятия, а также жестких сроков выхода продукта на рынок, эпизодическое управление требованиями становится все более и более неадекватным и неэффективным.

Для решения вышеперечисленных проблем, Rational шесть основных областей, профессиональных приёмов, которыми необходимо овладеть. Следующие приемы представлены последовательно, однако, в процессе эффективного управления требованиями они должны применяться постоянно

в различных целях. Здесь же они представлены в той последовательности, в которой желательно применять их на первой итерации нового проекта.

Приём 1: Анализ проблемы.

«Анализ проблемы», предлагается ряд методов, применяемых для достижения правильного понимания проблемы, решить которую призвана новая система ПО.

Анализ проблемы проводится с целью понять бизнес - задачи, целевые первоначальные потребности заинтересованных лиц и предложить высокоуровневые решения. Эти методы анализа находят “проблему за проблемой”.

При анализе проблемы соглашение достигается на основе реальных проблем, и если состав заинтересованных лиц определён. Границы первоначального решения и ограничения определяют на основе как технических, так и бизнес - возможностей. При необходимости, экономическое обоснование проекта проводится на основе анализа отдачи от инвестиций, которой можно ожидать от системы.

Приём 2: Понимание потребностей заинтересованных лиц.

«Понимание потребностей пользователя» - представление разнообразных приёмов выявления требований посредством общения с пользователями системы и другими заинтересованными лицами.

Требования имеют множество источников. Они могут исходить от любой из заинтересованных в результатах проекта сторон. Клиенты, партнеры, конечные пользователи и эксперты в области УТ являются источниками требований с одной стороны. С другой стороны – это Управление, члены проектной команды, контролирующие бизнес, а также регулирующие органы.

Важно знать, как определить, кто должен быть источником требований, как выявить требования по средствам общения с заинтересованными лицами.

Заинтересованными лицами в проекте называют людей, которые являются основным источником этих требований.

При разработке информационной системы для использования внутри компании, можно включить в команду разработчиков людей с опытом конечных пользователей и знаниями в бизнес - области. Очень часто в таком случае будут начинаться дискуссии на уровне бизнес – модели, а не на системном уровне. Если разрабатывается продукт для продажи на рынке, то можно набирать в команду больше маркетологов, чтобы лучше понимать потребности клиентов на этом рынке.

Методы выявления требований включают в себя интервью, мозговой штурм, концептуальное прототипирование, вопросники, конкурентный анализ. Результатом выявления требований является список запросов и потребностей, которые описаны текстуально и графически, и у которых между собой определены приоритеты.

Приём 3: Определение системы.

Определение системы - по сути, процесс преобразования и организации понимания потребностей заинтересованных лиц в осмысленное определение разрабатываемой системы. Для начала принимаются решения о том, что представляет собой требование, определяют формат документации, язык формализации, уровень требований, приоритет запросов, и оцениваются затраты, технические и управленческие риски, возможности. Отчасти эта деятельность может включать в себя прототипы и предварительный дизайн моделей непосредственно связанных с наиболее важными запросами заинтересованных лиц. Часто используют слово “описание” вместо “документ”, чтобы избежать предполагаемых ограничений, присущих совместному использованию последнего. Описанием может быть документ в письменном виде, электронный файл, изображение или любое другое представление, передающее краткие системные требования. Результатом определения системы является описание системы, как в текстовом, так и в

графическом виде. Так же описание системы может быть представлено следующим образом.

Приём 4: Управление границами системы.

Для успешного управления масштабом необходимо не только хорошо осознавать потребности, но и договариваться о том, что считать приемлемым для достижения успеха, а что сделать невозможно.

Масштаб проекта определяется набором требований, выдвигаемых к разрабатываемой системе. Управление рамки проекта по размеру имеющихся ресурсов (времени, людей и деньги) играет ключевую роль в управлении успешными проектами. Управление масштабом проекта является постоянной деятельностью, что требует многократных и дополнительных задач, дробящих проект на более мелкие более управляемые части.

Использование атрибутов требования, таких как приоритет, затраты на реализацию и риска в качестве основы для проведения переговоров о регистрации требования является особенно полезным методом для управления масштабом.

Приём 5: Уточнение определения системы.

Здесь необходимо организовать информацию о требованиях. Предлагается ряд методов уточнения определения системы до уровня детализации, пригодного для проведения проектирования и реализации, чтобы вся команда точно знала, какую систему она разрабатывает.

Необходимо и экономически выгодно вкладывать средства при более уточненном определении системы, когда согласовано на высоком уровне определение системы и достигнуто понимание первоначального объема работ.

Описания зачастую являются критически важными ссылочными материалами для проектных групп. Следует внимательно формировать описания. Распространенной ошибкой является мнение, что сложно строить сложное определение, особенно когда аудитория может быть не в состоянии

или не желает критически мыслить, и прийти к согласию. Это приводит к трудностям в объяснении цели системы для людей как внутри, так и за пределами проектной команды. Вместо этого, можно разработать различные описания для различных аудиторий. Этот артефакт будет включать в себя форматы детального описания на естественном языке, формализованного текста и графических описаний. После того, форматы описания установлены, его уточнение продолжается в течение всего жизненного цикла проекта.

Приём 6: Управление запросами на изменение.

В проектах внедрения методологии RUP сталкиваются с проблемой при описании требований на разработку отчётности. Из-за низкого качества постановок требований процесс разработки отчётности растягивается во времени, может сложиться парадоксальная ситуация, когда отсутствует понимание между аналитиками и разработчиками, и при этом никто не берёт на себя ответственность в разрешении сложившейся ситуации. Кроме того, на этом этапе система, включающая и среду для разработки отчётов, может находиться уже в стадии активной доработки

Независимо от того, насколько тщательно будут в дальнейшем определяться требования, они будут меняться. Учет меняющихся требований является мерой чувствительности команды к запросам заинтересованных лиц и оперативной гибкости – теми атрибутами команды, которые способствуют успешной реализации проекта. Изменение, по сути, не опасно, опасны неконтролируемые изменения.

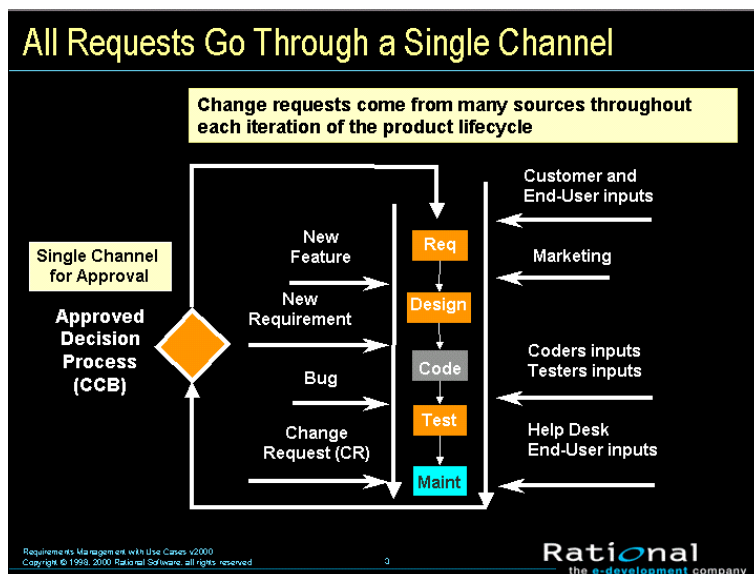


Рисунок 2-2 Процесс управления изменениями

Измененное требование означает, что больше или меньше времени придется потратить на реализацию свойств системы, и изменения в одного требования могут оказывать влияние на другие требования. Управление изменениями требований включает такие мероприятия, как создание базовой версии, отслеживание истории каждого требования, определение важных для прослеживания зависимостей и поддержание контроля версий. На Рисунок 2-2 показано, что важно установить контроль над изменениями, а так же процесс утверждения изменений, требующий рассмотрения всех предлагаемых изменения ответственными членами команды. Иногда это единый канал контроля, называемый Комитетом по контролю за изменениями (Change Control Board - CCB).

2.7 Важные понятия о требованиях

Чтобы применить навыки управления требованиями к проекту, все пользователи проекта должны иметь представление об определенных концепциях управления требованиями. Они включают в себя:

- Типы требований;
- Межфункциональные команды;

- Прослеживаемость (трассируемость);
- Многомерные атрибуты;
- История изменений.

2.7.1 Типы требований.

Чем больше и сложные системы, тем больше типов требований появится. Тип требования – это просто класс требования. Определяя типы требований, команды могут организовать большое число требований в значимые и более управляемые группы. Создание различных типов требований в проекте помогает членам команды классифицировать запросы на внесение изменений и общаться более четко.

Как правило, один тип требования может быть разделен или разложен на другие типы. Формулировка бизнес-правил и концепции может быть типом требований высокого уровня, из которых команда получает потребности пользователей, функций и требования к программному продукту. Сценарии использования и другие формы моделирования помогают разработать требования, которые могут быть разложены на требования к программному обеспечению и представлены моделями анализа и дизайна. Требования к тестированию вытекают из требований к программному обеспечению и распадаются на конкретные процедуры проведения тестирования. Когда существуют сотни, тысячи или даже десятки тысяч экземпляров требований в данном проекте, классификация по типам требований делает проект более управляемым.

2.7.2 Межфункциональные команды.

В отличие от таких процессов, как тестирование или моделирование приложений, которые могут решаться в рамках одной бизнес-группы, к управлению требованиями следует привлекать всех, кто может поделиться своим опытом в процессе разработки системы. Необходимо привлекать людей, которые представляют потребности клиентов и бизнеса. Менеджеры по разработке, менеджеры настройки, аналитики, системные инженеры, и

даже клиенты должны участвовать в процессе УТ. Группы управления требованиями должны также включать тех, кто создает системное решение - инженеров, архитекторов, разработчиков, программистов, технических писателей и других технических специалистов. Тестеры и другие представители контроля качества (КК), должны считаться важными членами команды.

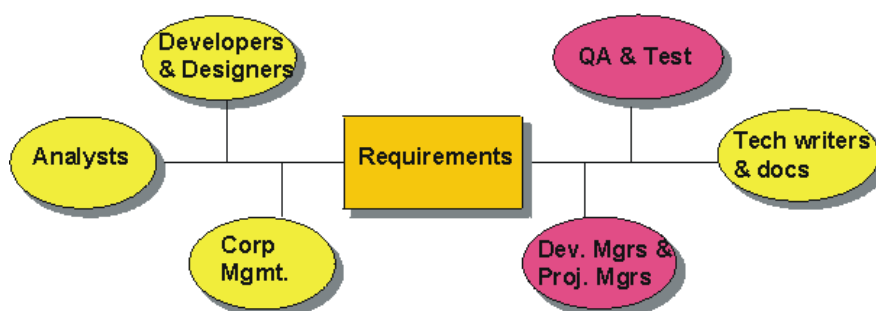


Рисунок 2-3 Межфункциональная команда по управлению требованиями

Ответственность за создание и сопровождение типов требований можно разделить по функциональным областям, что приводит к лучшему управлению большими проектами. Межфункциональный характер требований управление является одним из наиболее сложных аспектов дисциплины.

2.7.3 Трассировка.

Как указывается в определении типов требований, ни одно слово в требовании не стоит особняком. Запросы заинтересованных лиц, относящиеся к свойствам продукта, должны быть удовлетворены. Свойства продукта, связанные с отдельными требованиями, определяют особенности системы с точки зрения функционального и нефункционального поведения. Задачи тестирования связаны с требованиями верификации и валидации. Требования могут зависеть от других требований или же они могут быть взаимоисключающими. Для определения последствий изменений и

уверенности, что система соответствует ожиданиям, эти трассировочные зависимости должны быть понятны для всех команд, участвующих в работе по УТ, задокументированы и поддерживаться в актуальном состоянии. Хотя трассируемость является одной из наиболее сложно реализуемых концепций в управление требованиями, она весьма важна, даже неотъемлема в работе с изменениями. Установление четких типов требований и вовлечение межфункциональных команд может облегчить реализацию трассируемости и её поддержку. [9].

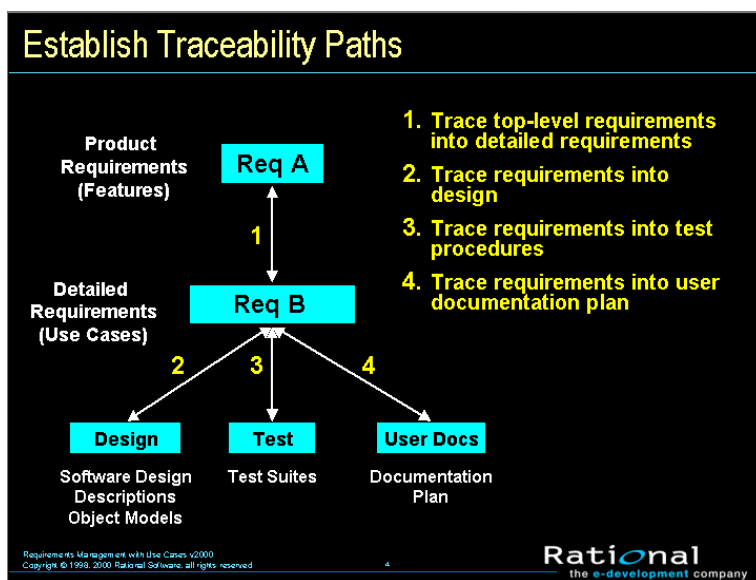


Рисунок 2-4 Трассировка требований

2.7.4 Многомерные атрибуты.

Каждый тип требования имеет атрибуты, и каждое отдельное требование имеет различные значения атрибута. Например, требованиям могут быть присвоены приоритеты, определенные источником и обоснованные, учитывающие степень сложности выполнения или связанные с конкретной итерацией системы. Для иллюстрации на Рисунок 2-5 изображены атрибуты для Типа Требования “Свойство” в инструменте управления требованиями Rational RequisitePro. Как следует из названия

экрана¹, тип требования и атрибуты для каждого типа определяются для всего проекта, обеспечивая согласованное использование в рамках всей группы.

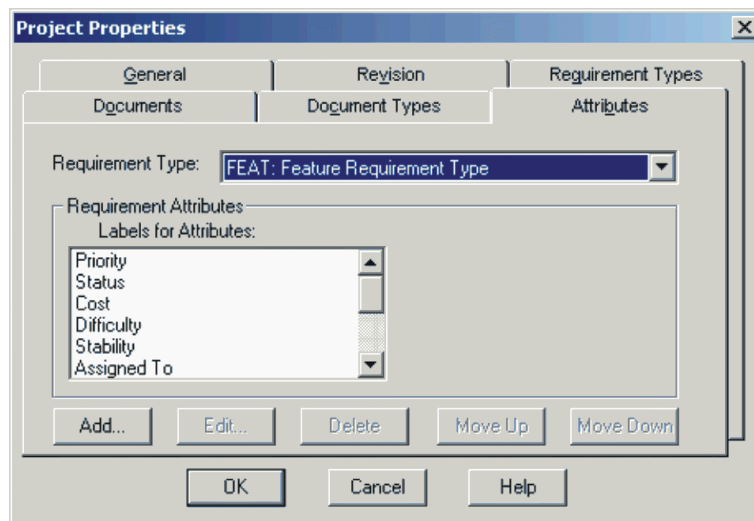


Рисунок 2-5 Определение Атрибутов требований для Свойств

На Рисунок 2-6 приведен пример функциональных требований, отображающихся для конкретного проекта RequisitePro. Даже без отображения всего текста по каждому требованию, можно многое узнать о каждом требовании из значений его атрибутов. В этом случае, его приоритет и сложность реализации (установленных различными членами команды) помогут команде оценить рамки, масштаб проекта на основе имеющихся ресурсов и времени, с учетом как приоритетов заинтересованных лиц и очень грубой оценки трудозатрат, отраженной в значении атрибута сложности реализации. Детализируя атрибуты типов требований “приоритет” и “сложность реализации”, можно вводить более конкретные значения (например, расчетное время, количество строк кода, и так далее), на основе которых в дальнейшем уточняются рамки проекта. Многомерный аспект требования, которое включает в состав различные типы требований - каждый со своими собственными атрибутами - имеет важнейшее значение для

¹ Project Properties – Свойства Проекта.

организации большого количества требований и управления рамками всего проекта.

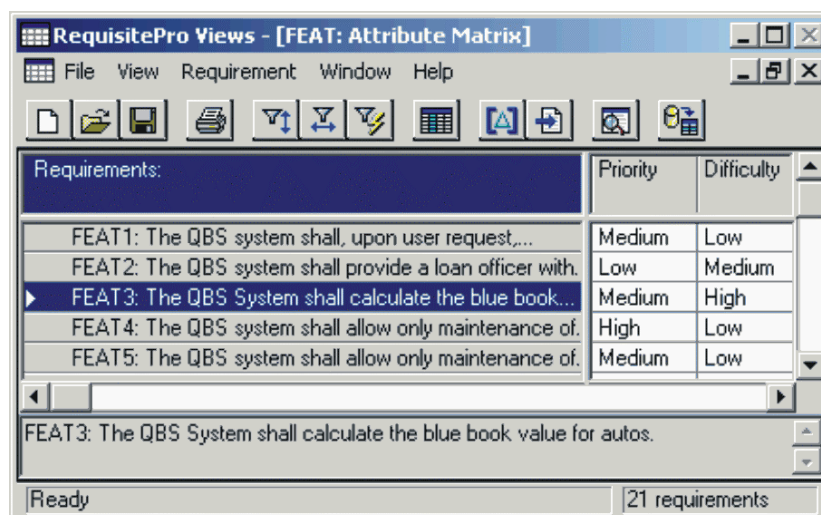


Рисунок 2-6 Настройка значений атрибутов для каждого требования

2.7.5 История изменений.

Как отдельно взятые требования, так и группы требования имеют свою историю изменений, которая становится значимой, важной с течением времени. Перемены неизбежны, и желательно идти в ногу с постоянно меняющимися условиями и новыми технологиями. Запись версии требований проекта позволит руководителям групп определить главные причины изменений проекта.

Понимание того, что совокупность требований может быть связана с определенной версией разрабатываемого программного обеспечения, позволяет управлять изменениями постепенно, снижая риски и повышая вероятность успешно завершать этапы работ проекта. Пока отдельные требования развиваются, важно понимать их историю изменений: что изменилось, почему, когда, и даже кто это решение об изменении принял.

2.8 Управление требованиями на практике.

Управление требованиями основывается на вышеизложенных ключевых навыках и концепциях для успешного выявления и устранения проблем проекта.

Чтобы построить систему, которая действительно отвечает потребностям клиентов, команде проекта необходимо определить проблему, решать которую будет разрабатываемое (или внедряемое) ПО. Далее, команда должна определить заинтересованные стороны, потребности которых вызваны бизнесом или конечными пользователями, описаны и приоритизированы. Из этого набора ожиданий высокого уровня или потребностей формируется набор решений или функций системы, который должен быть обязательно согласован.

Подробные же требования к программному обеспечению должны быть написаны в форме, которая понимается как заказчиками, так и разработчиками. Считается, что использование описания требований к программному обеспечению на языке клиента является наиболее эффективным в стремлении к пониманию и согласию. Эти подробные требования затем используются в качестве материала при составлении спецификаций на проектирование системы, а также для планов испытания и процедур, необходимые для осуществления валидации и верификации. Требования к программному обеспечению также ложатся в основу постановок задач на проектирование и планирования.

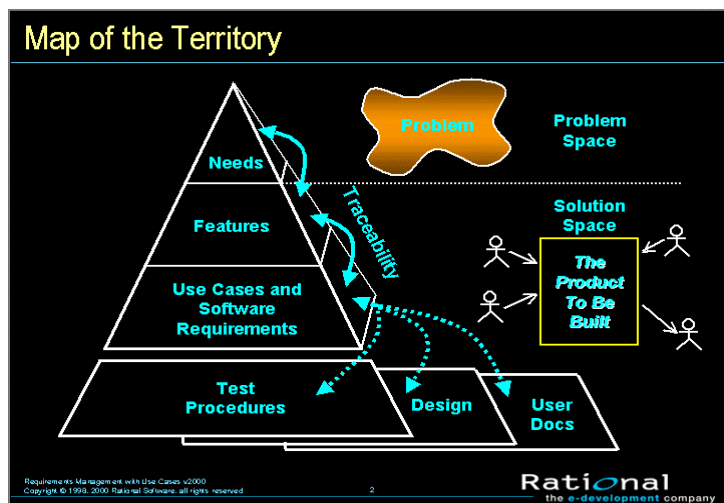


Рисунок 2-7 Обзор структуры управления требованиями

Для облегчения работ по проекту команде необходимо:

- Согласование общей лексики проекта, основных определений;
- Разработка концепции системы, которая описывает проблему, которую необходимо решить в рамках проекта, а также в качестве основной функции разрабатываемой системы;
- Выявить потребности заинтересованных лиц, по крайней мере, в пяти важных областях: функциональность, удобство использования, надежность, производительность и возможности поддержки;
- Определить, какие типы требований надо будет использовать;
- Выбрать атрибуты и их значения для каждого типа требования;
- Выбрать формат, в котором будут описаны требования;
- Определить членов команды, которые могут стать авторами либо будут помогать в составлении или просто просматривать одного или несколько типов требований;
- Решить, связи между какими требованиями необходимо отслеживать;
- Разработать процедуры предложения, анализа и внесения изменений в требования;
- Разработать механизм для отслеживания истории изменения требований;

- Создать отчёты о состоянии требований и их выполнении для членов команды и управления.

Эти основные этапы управления требованиями не зависят от отрасли, методологии разработки, инструментов или требований. Они настолько гибкие, что позволяют эффективно управлять требованиями в самых различных средах разработки приложений.

2.9 Управление требованиями: потоки работ.

Описанный ниже подход к УТ предусматривает 6 детализированных рабочих процессов, которые применяются к каждому из ключевых приемов управления требованиями и могут быть применены в любой области.

Следующие схемы потоков работ описаны в RUP [6] в дисциплине “Требования: поток работ”. Эти процессы выражаются в терминах ролей, видов деятельности и артефактов (вход или выход) как показано на Рисунок 2-8. Каждый процесс сопровождается кратким текстовым описанием.

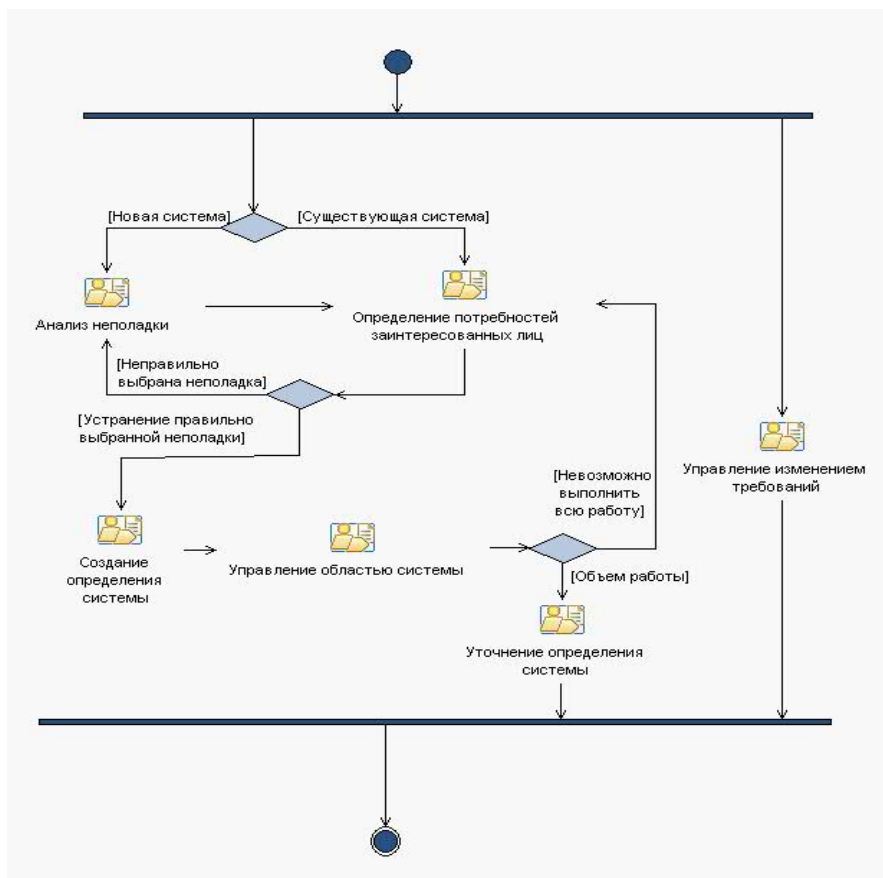


Рисунок 2-8 Поток работ Управления требованиями в RUP

2.9.1 Поток работ: Анализ проблемы.

В задаче анализа основным мероприятием является разработка концепции проекта. Результатом анализа проблемы должен стать документ-концепция, определяющая на высоком уровне систему, которую необходимо разработать, с точек зрения пользователей или клиентов. Концепция отражает первоначальные требования, определяются ключевые функции системы, необходимые для решения критичных проблем и удовлетворения потребностей заинтересованных лиц. Системный аналитик играет главную роль в этом процессе. Системный аналитик должен иметь опыт работы в данной области и понимание основной проблемы, должен описать процесс, который по его или ее мнению решить эту проблему. Требуется активное

вовлечение различных заинтересованных лиц проекта, и все соответствующие просьбы заинтересованных лиц должны быть рассмотрены.

Чтобы начать мероприятия по Управлению зависимостями, свойствам системы должны быть назначены такие атрибуты, как обоснование, относительная стоимость, или приоритет, и источник запроса. В то время как разрабатывается концепция, аналитик идентифицирует пользователей и системы (актеров) возможных вариантов использования. Актеры - главный элемент модели прецедентов, которые будут определять функциональные и нефункциональные технические требования системы.

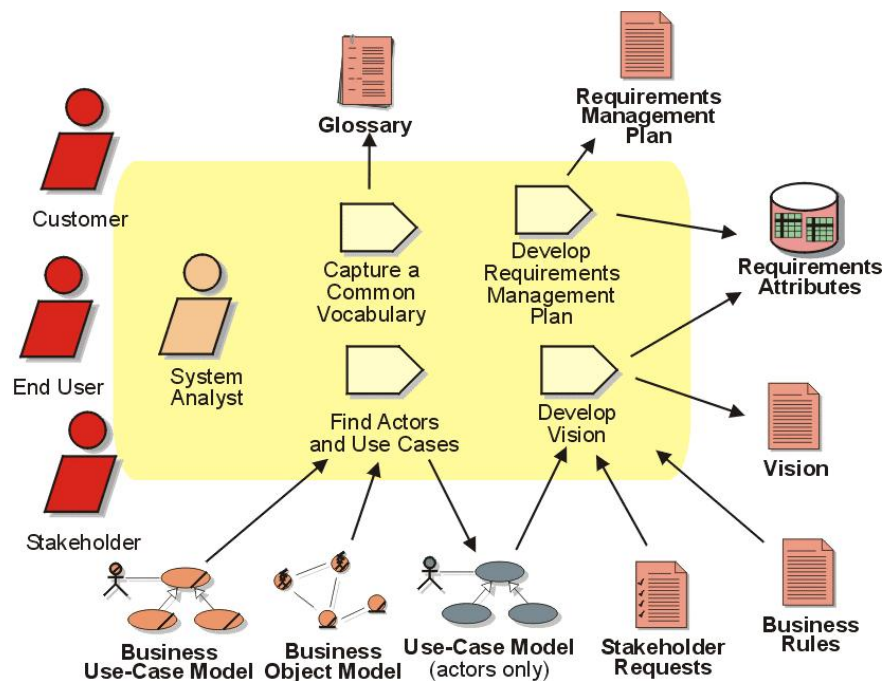


Рисунок 2-9 Анализ проблемы

Системный аналитик из команды разработчиков может помочь заинтересованным сторонам первоначально описать проблему, которую они хотят решить. Важно достичь соглашения о краткое изложение предполагаемой проблемы. Ключевые элементы **постановки задачи** приведены в следующей таблице:

Таблица 1 Постановка задачи.

Проблема в:	определить проблемы
-------------	---------------------

Влияет на:	список заинтересованных лиц, затронутых проблемой определения проблемы
Влияние заключается в:	описать влияние проблемы
Требования на успешное решение:	перечислить некоторые основные преимущества успешного решения

Постановка задачи кратко объясняет цель проекта. Анализ проблемы стимулирует дальнейшее исследование всех запросов заинтересованных лиц и первоначальной бизнес-задачи. Параллельно с разработкой постановки задачи, команда должна также составить глоссарий для отслеживания наиболее часто используемых терминов и согласования их определения.

При анализе проблемы также определяют основные субъекты системы. Актерами являются пользователи системы или любая другая система, с которой будет осуществляться обмен информацией. На данном этапе анализа проблемы следует кратко определить ряд очевидных последовательностей будущих взаимодействий актеров с системой. Описания должны быть ориентированы на бизнес-процесс, а не поведение системы. Например, программа бюджетирования может позволить одному типу актеров “Создать ведомственный бюджет”, а другой актер сможет “Консолидировать ведомственный бюджет”. Системный аналитик в дальнейшем может разделить их на дополнительные прецеденты, которые наиболее соответствуют осознанию конкретного поведения системы. Например, “Создание ведомственный бюджета” может быть разделен на прецеденты “Импорт информационной таблицы ” и “ Создание просмотра бюджета”.

2.9.2 Поток работ: Понимание потребностей заинтересованных лиц.

Если первоначальный анализ проблемы оправдал инвестиции в разработку системы, следует начинать мероприятия по Пониманию потребностей заинтересованных лиц. Основным направлением деятельности становится выявление запросов заинтересованных лиц. Результатом будет

являться набор приоритезованных потребностей заинтересованных лиц и свойств системы, которые позволят уточнить Концепцию и обеспечить лучшее понимание атрибутов требований. Кроме того, в ходе этого процесса можно приступить к обсуждению системы в терминах вариантов использования и актеров. Еще одним важным результатом является обновленный глоссарий терминов.

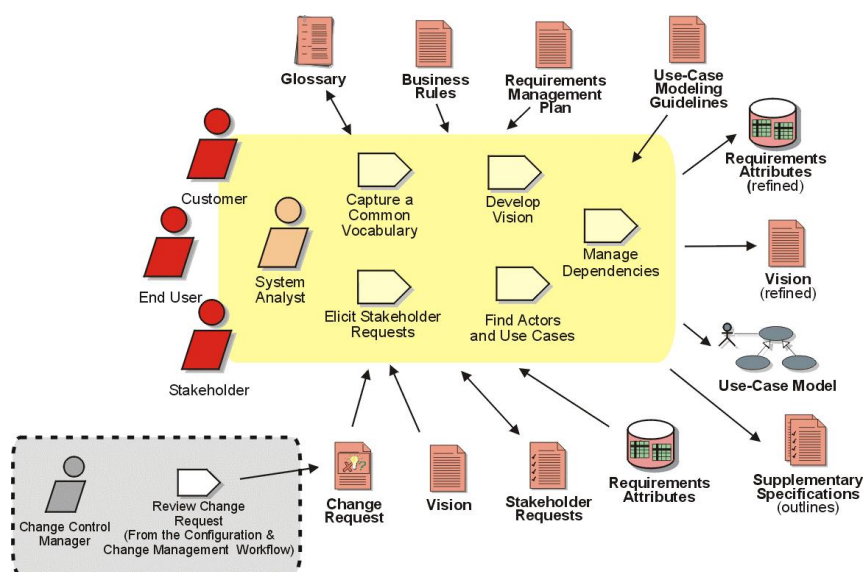


Рисунок 2-10 Понимание потребностей заинтересованных лиц

Системный аналитик и ключевые заинтересованные лица определяют дополнительные заинтересованные стороны, выявляют их запросы и определяют основные потребности и необходимые им свойства системы посредством интервью, семинаров, раскадровки, бизнес-процессов, вариантов использования и других методов. Один или несколько системных аналитиков могут помочь в этом. Такие методы выявления требований являются одними из самых полезных и эффективных. Процесс включает в себя пользователей, справочную службу персонала, владельцев бизнеса, тестеров и др., заинтересованных в успешном завершении данного проекта. Запросы заинтересованных лиц часто неоднозначны, совпадают и иногда несовместимы. В дополнение к официальным результатам выявления, запросы заинтересованных лиц могут быть выражены в хорошо

отформатированных документах, запросах к базе данных или электронных письмах и групповых обсуждениях. Системный аналитик регистрирует, распределяет по категориям и приоритезирует выявленные основные потребности заинтересованных лиц.

На основе лучшего понимания потребностей заинтересованных лиц системные аналитики команды разработчиков уточняют Концепцию, обращая особое внимание на разработку “соглашения о моделировании”. В двух-трех предложениях это соглашение отражает реальную пользу проекта. Соглашение должно включать предполагаемых пользователей, проблемы, которые решает система, выгоды, которые она предоставляет, и конкурирующие продукты, которые она заменяет. Все члены команды должны понимать цель проекта. Системные аналитики также обновляют глоссарий, оказывая содействие общему пониманию терминов.

Comment [I.V.1]: Product position statement

2.9.3 Поток работ: Определение системы.

После первых двух мероприятий по анализу проблемы и пониманию потребности заинтересованных лиц созданы первые итерации основных определений системы, в том числе свойств, указанных в Концепции, первый вариант модели прецедентов, а также первоначальные атрибуты требований. Следующий процесс Определения системы завершает описание требований к системе высокого уровня с усовершенствованными определениями свойств, помимо новых действующих лиц, вариантов использования, а также дополнительных спецификаций.

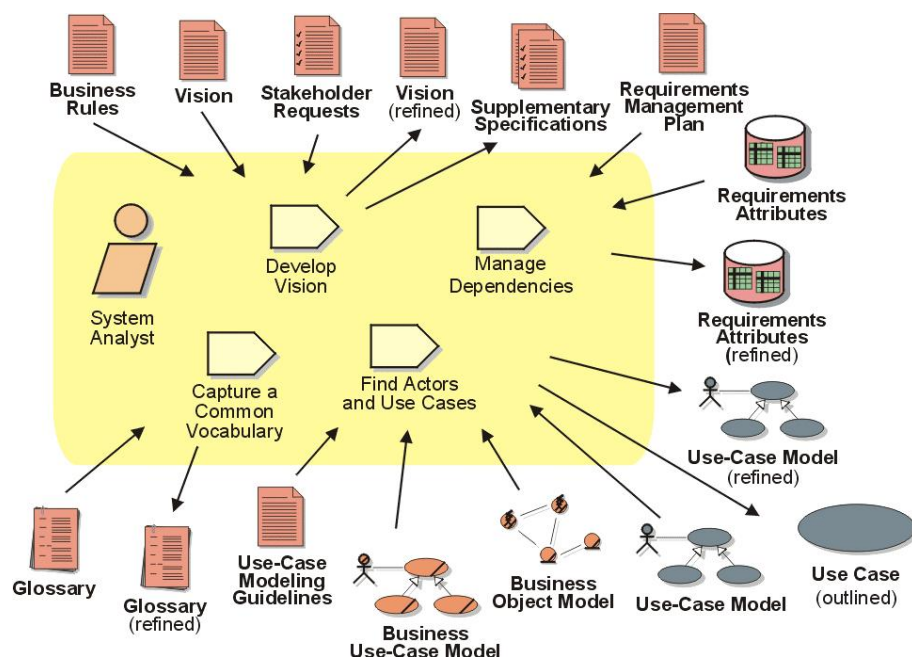


Рисунок 2-11 Определение системы

Глоссарий обновляется, чтобы отражать актуальные определения, используемый для описания свойств и требований из моделей прецедентов и дополнительных спецификаций.

Системный аналитик использует набор свойств из обновленной Концепции для получения и описания прецедентов, отражающих мнение пользователей об ожидаемом поведении системы. Модель прецедентов - это договоренность между заказчиком, пользователями и разработчиками системы о том, как выбранные свойства будут функционировать в системе. Это помогает установить реалистичные ожидания и цели для разработчиков системы и помогает клиентам и пользователям проверить, что система будет соответствовать этим ожиданиям.

Некоторые системные требования не вписываются в модель прецедентов. Системный аналитик описывает их в дополнительных спецификациях. Многие нефункциональные требования, такие как требования удобства, надежности, производительности и возможности поддержки часто так и остаются в спецификациях. Следует отметить, что

многие нефункциональные требования такого рода характеризуются отдельным вариантом использования. Разработчикам вариантов использования лучше включать эти требования в спецификации моделей прецедентов, оставляя дополнительные спецификации для общесистемных нефункциональных требований.

В ходе того процесса системный аналитик создает и устанавливает атрибуты для дополнительных требований (такие, как приоритет и связанных с требованиями вариантов использования). Кроме того, системный аналитик добавляет и обновляет значения атрибутов для первоначальных и новых вариантов использования.

Наконец, системный аналитик управляет зависимостями путем трассирования важных потребностей пользователей и критических свойств со связанными вариантами использования и требованиями, приведенными в дополнительных спецификациях.

2.9.4 Поток работ: Управление границами системы.

После определения требований на уровне свойств, описания большинства актеров, прецедентов и других требований, указанных в дополнительных спецификациях, системный аналитик собирает и присваивать значения с максимально возможной точностью атрибутам требований таким, как приоритет, усилия, затраты и риски. Это позволяет определить первоначальный объем работ для выпуска системы, а также позволит определить архитектору варианты использования, связанные с архитектурой системы.

План итераций, разрабатываемый в ходе проекта и управления разработкой, разрабатывается на основе результатов процесса Управления масштабом системы. Также известный как план разработки, план итераций определяет количество и частоту итераций, запланированных для выпуска системы. Элементы объема работ с самым высоким риском должно быть запланированы на ранних итерациях.

Важными результатами рабочего процесса Управления масштабом являются первоначальный вариант документа архитектуры программного обеспечения и пересмотренная Концепция, которая отражает более глубокое понимание функциональных возможностей системы и ресурсов проекта аналитиками и ключевыми заинтересованными лицами.

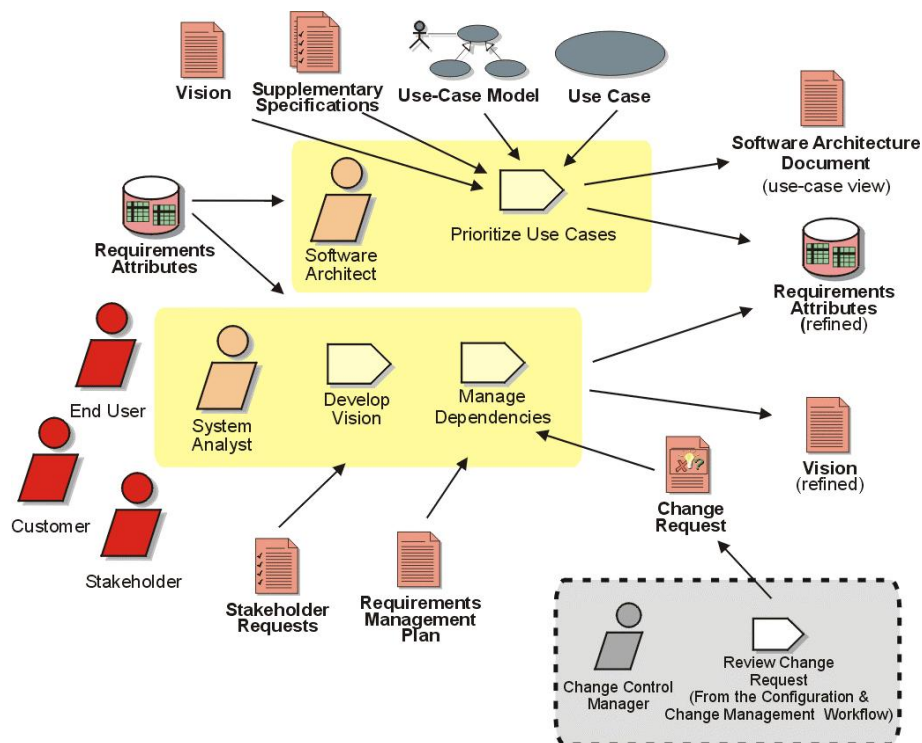


Рисунок 2-12 Управление границами системы

Архитекторы приоритезируют варианты использования для снятия рисков, выделяя наиболее важные для реализации системы. Системные аналитики управляют зависимостями, уточнив атрибуты требований для свойств в Концепции. Они также уточняют требования в вариантах использования и дополнительных спецификациях. Системные аналитики обеспечивают надлежащие трассировки требований, созданных для отслеживания потребностей заинтересованных лиц, свойств, требований из вариантов использования, а также требования дополнительных спецификаций.

Этот шаг является наиболее важным для всего проекта. Впервые объем знаний о предлагаемой системе позволяет зафиксировать требования, проектные ресурсы и сроки исполнения. В то же время, следует понимать, что эти требования будут меняться, а глубина знаний - увеличиваться. Если управление зависимостями осуществлялось на предыдущих трех этапах, то на этом шаге будет легче работать с будущими изменениями.

За время существования проекта, при изменении ситуации и условий, этот процесс вновь повторяется столько раз, сколько системным аналитикам понадобится времени на пересмотр объема работ по проекту и Концепции с основными заинтересованными сторонами. Управление масштабом проекта в рамках имеющихся средств успешно только тогда, когда заинтересованные стороны и члены команды разработчиков оценивают этот этап, как естественное развитие - не лишнюю необходимость отвечать ожиданиям пользователей или попытку выманить у организации больше времени и денег. Этот процесс необходимо повторять на основных вехах проекта, чтобы оценить, по-новому взглянуть на систему и ее проблемы, требующие изменения масштаба. Хотя утвержденные требования, бюджеты и сроки трудно изменить, глубокое понимание приоритетных прецедентов, дополнительных спецификаций и ранние итерации системы неизбежно приведет к пересмотру объемов работ.

Опять же, важно, чтобы проектная группа занималась обычным управлением масштабом, не переходя к стадии уточнения, в течение всего жизненного цикла проекта, пока происходят изменения. Представитель заинтересованных лиц должен понимать и верить, что их приоритезация и интересы принимаются всерьез. С течением времени системные требования уточняются, и только важные потребности остаются без изменений. Если рамки эффективных навыков управления не были установлены, проект может быть обречен - безнадежно увеличенный объем работ проекта неумолимо приблизится к перерасходу средств и потере времени.

2.9.5 Поток работ: Уточнение определения системы.

Процесс Уточнения определения системы предполагает изложение в общих чертах вариантов использования на системном уровне и короткое описание актеров. При управлении рамками проекта уточнились приоритеты свойств в Концепции, и на данном этапе они могут быть реализованы за довольно плотные сроки и бюджет. Результатом этого процесса является более глубокое понимание функциональных возможностей системы, выражается в виде подробных вариантов использования, пересмотренных дополнительных спецификаций и ранней версии самой системы.

Конечно, не все системы будут иметь интерфейсы пользователя, и не на всех ранних итерациях будет присутствовать элементы GUI². В некоторых случаях в ранние версии системы включают прототипы, модели и раскраски.

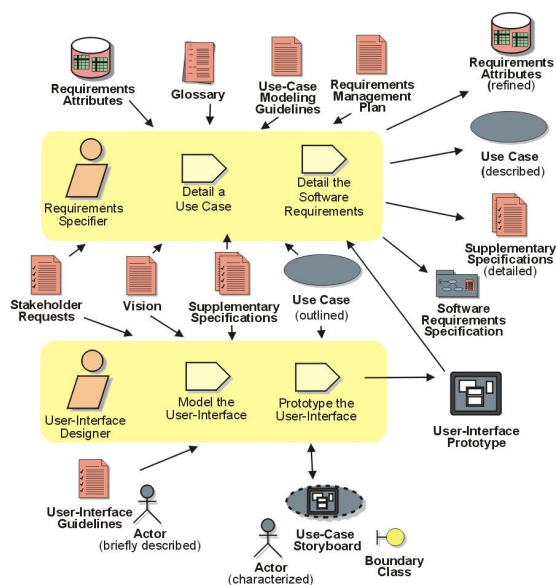


Рисунок 2-13 Уточнение определения системы

Составитель спецификаций вариантов использования подробно описывает поток событий, пред- и постусловия, а также другие текстовые

² GUI – Graphical user interface – графический интерфейс пользователя, программа, позволяющая осуществлять визуализацию данных.

свойства каждого варианта использования. Чтобы свести к минимуму сложность и повысить читаемость, сохранить все текстовые описания прецедентов, рекомендуется использовать стандартный формат документов или шаблоны спецификаций вариантов использования. Создание хорошо продуманных шаблонов спецификаций прецедентов имеет решающее значение для качества системы. Эта разработка спецификаций требует глубокого понимания потребностей заинтересованных лиц и свойств системы, связанных с прецедентами. Желательно дополнительное участие нескольких членов команды проекта (например, разработчиков ПО) в создании прецедентов.

Параллельно с этим, составитель спецификаций вариантов использования прорабатывает дополнительные спецификации с дополнительными требованиями, которые не вошли в варианты использования.

Разработчик интерфейса пользователя моделирует и создает прототип пользовательского интерфейса системы. Эта работа тесно связана с постепенным развитием вариантов использования.

Составитель спецификаций прецедентов и системный аналитик пересматривают сложность, затраты, риски и другие значения атрибутов для каждого требования.

Результаты этого уточнения определения системы позволяют более обоснованно контролировать и управлять рамками проекта. После того как получено больше информации о будущей системе, можно изменять, корректировать приоритеты. Т.к. на данном этапе существует большая вероятность того, что объем работ по выпуску системы необходимо будет пересматривать и уточнять.

2.9.6 Поток работ: Управление запросами на изменение.

Так как появление изменений неизбежно, то процесс управления изменяющимися требованиями необходимо применять постоянно в течение

всего срока проекта. В результате этого процесса могут привести к изменению каждого документа, который, в свою очередь, требует эффективного взаимодействия всех членов проектной группы и заинтересованных лиц.

Изменения в требованиях естественно влияют на модели, построенные в результате процессов анализа и проектирования системы. Изменения требований также влияют на тестирование, направленное на проверку надлежащего выполнения требований. Трассировка зависимостей, выявленных в процессе управления зависимостями, помогает оценить влияние воздействий от изменений в требованиях.

Важным понятием процесса управления изменяющимися требованиями является отслеживание истории требований. Определив характер и обоснование изменения требований, рецензенты (кто-либо из команды разработчиков, работа которых зависит от изменений) получают информацию, необходимую для реагирования на изменения должным образом.

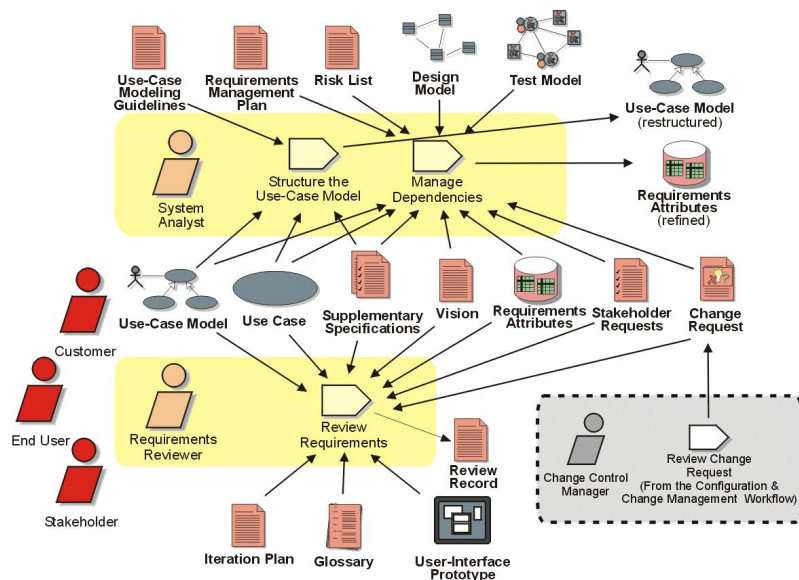


Рисунок 2-14 Управление запросами на изменение

Запросы на внесение изменений в требования могут быть выдвинуты любым заинтересованным лицом или членов проектной группы по многим

различным причинам. Все запросы на изменение (CRs-Change Requests), в том числе изменения в требованиях или расширения запросов, а также дефекты, должны контролироваться процессом управления запросами на изменения (CRM-Change Request Management). Как минимум, этот процесс включает в себя регистрацию и отслеживание запросов в централизованной базе данных и обязательное рассмотрение центральным проверяющим органом.

Системный аналитик должен координировать мероприятия проверки (желательно) группой контроля за внесенными изменениями (Change Control Board - CCB), сбора и анализа всех запросов на изменения и классифицировать их как:

- недостатки в реализации, которые не влияют на требования;
- изменения в существующих требованиях того или иного типа;
- запросы на модификацию.

После классификации предлагаемым изменениям в требованиях присваиваются атрибуты и значения, как описано в других процессах управления требованиями.

При рассмотрении запросов на изменения, системный аналитик представляет требования с предлагаемыми приоритетами для изменения в CCB, состоящий из представителей заинтересованных лиц и членов команды проекта. Объем изменений, который превышает имеющиеся ресурсы, должен быть отклонен или доведен до сведения представителей заинтересованных лиц, которые уполномочены утвердить необходимые изменения сроков и финансовых обязательств.

CCB одобряет или отклоняет изменения требований.

Системный аналитик сообщает об изменениях требований разработчикам требований или вносит изменения непосредственно в требования в Концепцию, варианты использования, документы дополнительных спецификаций и другие артефакты с требованиями.

Редакторы требований (разработчики, тестировщики, менеджеры и другие члены команды проекта) оценивают воздействие изменений в требованиях на их работу с помощью истории изменений требований. Наконец, они осуществляют изменения и вносят соответствующие поправки в соответствующие требования, авторами которых являются редакторы.

2.10 Средства УТ. Лидеры и остальные.

Существующие на данный момент средства УТ в большей или меньшей степени удовлетворяют следующим требованиям:

- Формирование структуры требований
- Фиксирование исходных и возникающих требований
- Визуальное оформление требований
- Автоматическое связывание требований между собой
- Автоматическое связывание требований с задачами на работы
- Контроль хода выполнения задач
- Определение необходимых типов требований и атрибутов для

них

- Генерация актуального технического задания
- Интеграция со средой разработки
- Учет истории и авторства изменений требований
- Анализы влияния изменений (change impact)
- Сбор, обработка и использование статистики требований
- Средства ведения форумов и дискуссий по требованиям
- Удалённая работа с требованиями

Наиболее полной функциональностью обладает “большая тройка” СУТ:

- Borland CaliberRM (Enterprise Software Requirements Management System) от \$2000 за лицензию.

- IBM Rational RequisitePro (Requirements Management Tool) \$2750-\$5350

- Telelogic DOORS (Requirements Management for Advanced Systems and Software Development) \$2500-\$8000.

Альтернативными системами являются: Optimal Trace, Compuware (Requirements definition and management); Visual Information Portal (VIP), Envision (Enterprise Management Control Center), \$3800-\$6700; Sparx System's RaQuest; Analyst Real System 7.0 (ARTS); Vitech CORE Product Suite; Cradle REQ; ViewSet PACE Requirements Management; Serena Dimensions RM; UGS Teamcenter SE and RE; \$50-\$1000.

Для управления требованиями компания Rational Software разработала инструмент Rational RequisitePro. Он предлагается как в виде отдельного продукта, так и в качестве ключевого компонента пакета Rational Suite AnalystStudio, который ниже будет рассмотрен более подробно. Rational Suite последовательно организует инструменты и процессы для поддержки различных функций проектной группы, объединенные Rational в виде отдельных "студий". Эти функции включают в себя управление проектом, управление требованиями и их анализ, разработку ПО, управление содержанием и тестирование системы. В основе Rational Suite лежит общая для всех студий "объединяющая платформа группы", в которую входят Rational Unified Process, управление требованиями, управление тестированием, отслеживание дефектов и изменений, а также управление конфигурацией:

- IBM Rational ClearQuest в управлении изменениями,
- Rational Rose (Rational XDE) создание и хранение описаний сценариев использования,
- TestManager в тестировании,
- IBM Rational ClearCase - инструмент версионного хранения и управления конфигурациями,

- SoDA – инструмент автоматического формирования различных документов, содержащих требования к ПО, отчётов о ходе работ с требованиями.

RequisitePro может быть также интегрирован с MS Project. Это позволяет установить и отслеживать связь требований и задач MS Project, которые предполагают реализацию этих требований. (Рисунок 2-15).

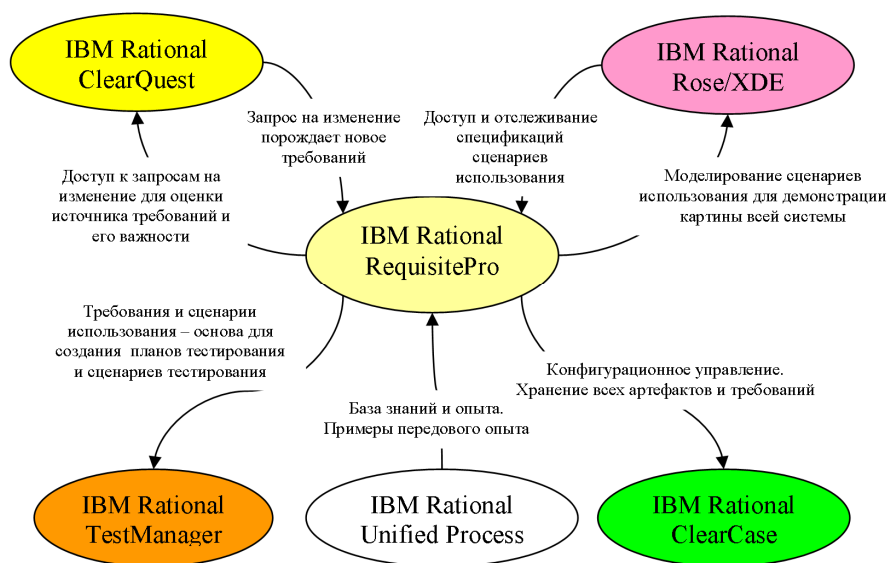


Рисунок 2-15

RequisitePro позволяет фиксировать различные типы требований, формировать для них необходимые атрибуты, осуществлять их структуризацию и трассировку.

В RequisitePro требования хранятся в базе данных. Для каждого требования хранится полная история его изменений и расширяемый набор атрибутов. RequisitePro предоставляет удобный графический интерфейс для ввода требований и работы с ними, который позволяет легко выбирать и сортировать требования по различным критериям. (Рисунок 2-16).

Requirements:	Priority	Status	Difficulty	Stability	Type	Phase
BLR16.8.3.2: Если роумерская сессия на...	Medium	Approved	Medium	Medium		
BLR16.8.3.3: Дата отбытия, в этом случае...	Medium	Approved	Medium	Medium		
BLR16.8.4: Изменение даты и времени...	Medium	Approved	Medium	Medium		
BLR16.8.4.1: Должна существовать...	Medium	Approved	Medium	Medium		
BLR16.8.4.2: Если статус номера изменен, то...	Medium	Approved	Medium	Medium		
BLR16.8.4.3: Если после изменения статуса...	Medium	Approved	Medium	Medium		
BLR16.8.5: Возвращение нашего роумера	Medium	Approved	Medium	Medium		
BLR16.8.5.1: Заказ доступен на стадии сессии	Medium	Approved	Medium	Medium		
BLR16.8.5.2: При выполнении заказа...	Medium	Approved	Medium	Medium		
BLR16.8.5.3: При выполнении заказа...	Medium	Approved	Medium	Medium		
BLR16.8.5.4: Цель заказа ? отмена статуса...	Medium	Approved	Medium	Medium		
BLR16.8.5.5: После выполнения заказа...	Medium	Approved	Medium	Medium		
BLR16.9: Подключение в городе роуминга	Medium	Approved	Medium	Medium		
BLR16.9.1: Заказ доступен на стадии...	Medium	Approved	Medium	Medium		
BLR16.9.2: При выполнении заказа в поле...	Medium	Approved	Medium	Medium		
BLR16.10: Функциональное назначение...	Medium	Approved	Medium	Medium		
BLR16.10.1: Подключение в московской сети	Medium	Approved	Medium	Medium		
BLR16.10.1.1: Заказ, формирующий и...	Medium	Approved	Medium	Medium		
BLR16.10.1.2: Подключение производится на...	Medium	Approved	Medium	Medium		
BLR16.10.1.3: Подключение гостя ? роумера...	Medium	Approved	Medium	Medium		
BLR16.10.1.4: В случае подключения с пин ?...	Medium	Approved	Medium	Medium		
BLR16.10.1.5: При получении ответа с...	Medium	Approved	Medium	Medium		
BLR16.10.2: Подключение на постоянный...	Medium	Approved	Medium	Medium		

BLR16.8.5.1: Заказ доступен на стадии сессии ?оформлен?, ?подключен?

Ready 2647 requirements

Рисунок 2-16

IBM Rational RequisiteWeb позволяет работать с документами и базой данных RequisitePro через Интернет. Это особенно удобно при организации работы регионально удаленных команд, а также при сборе требований из множества филиалов основной разрабатывающей компании.

Для согласования требований между участниками разработки и другими заинтересованными лицами RequisitePro позволяет организовывать дискуссии. Каждый участник дискуссии может высказаться как непосредственно по теме дискуссии, так и по поводу позиции других участников дискуссии.

Участие в дискуссии возможно как непосредственно из RequisitePro, так и с использованием электронной почты. В последнем случае высказывания участника дискуссии рассылаются всем остальным участникам дискуссии по электронной почте. Их ответы (по электронной почте) автоматически заносятся в базу данных RequisitePro. Такой механизм проведения дискуссий позволяет привлекать к участию в них не только непосредственных разработчиков, но и других заинтересованных лиц внутри и вне компании-разработчика, например, будущих пользователей системы.

Это позволяет активно сотрудничать с заказчиком в течение всего времени разработки ПС.

При желании можно настроить RequisitePro таким образом, чтобы в случае изменения требований сообщения об этом рассылались всем заинтересованным лицам.

RequisitePro позволяет использовать для работы с требованиями различные типы документов: описания сценариев использования, нефункциональные требования, концепцию (Vision) и другие. При необходимости можно создавать собственные типы документов, требований и определять набор атрибутов для описания этих требований (типовыми атрибутами требований являются, например, приоритет, сложность реализации и статус). Можно, например, разрабатывать шаблоны документов, соответствующих требованиям ГОСТ 34 или ГОСТ 19.

RequisitePro поддерживает групповую работу с требованиями. Разработчики могут быть разбиты на несколько групп, и для каждой группы могут быть назначены различные права на создание, редактирование и просмотр требований.

Для хранения требований RequisitePro может использовать различные СУБД в зависимости от объема требований и числа участников разработки: MS Access, MS SQL Server или Oracle. Это обеспечивает возможность масштабирования от использования RequisitePro в группах из нескольких разработчиков до самых крупных проектов.

2.11 Выводы

В Главе 1 “Проблемы в управлении требованиями” представлены основные методы управления требованиями и проанализированы основные проблемы, возникающие при внедрении данной дисциплины в процесс разработки ПО. На основе опыта передовых IT-компаний, информации интернет – источников предложена инструментальная поддержка процесса УТ. Предложено решение проблем УТ в совместном проекте компаний ОАО

“ИБС” и ОАО “Газпром” по созданию Автоматизированной системы формирования консолидированной отчётности (Проект “АСКО”).

Итак, почему столь большое внимание уделяется управлению требованиями?

Во-первых, цель любого ИТ-проекта - создание за отведённое время и бюджет качественного ПО, удовлетворяющего потребностям Заказчика. А если эти условия не соблюдаются, необходимо совершенствовать методы управления проектами и в частности, управление требованиями, т.к.:

- Ошибки в определении и изменении самих требований наиболее распространённое явление в проекте.
- Увеличение стоимости исправления ошибок и времени на реализацию изменений приводят к затратам, составляющим 25-40% бюджета проекта разработки в целом.

Во-вторых, практика управления требованиями, обобщённая в Главе 1, воплотила коллективный опыт тысяч людей, и отлично учитывает мнения многих людей, потративших годы работы с клиентами в области управления требованиями.

Наконец, за последнее десятилетие Rational Software сохраняет позицию лидера в предоставлении эффективных решений в области развития программного обеспечения, успешно решает задачу управления требованиями и выпустила инструмент для автоматизации этой сложной задачи. “Всеобъемлющие хронические проблемы управления требованиями разрешимы. А это, в конечном счете, может быть лучшим поводом, чтобы начать практику передового опыта в области управления требованиями уже сегодня.”[7].

3 Глава 2 “ Управление требованиями в АСКО ”.

Целью данной главы является определение требований верхнего уровня к системе управления требованиями (СУТ) в проекте «АСКО» с точки зрения потребностей пользователя.

Исследование области проблемы и постановка задачи см. Приложение 1 “Постановка задачи на проектирование СУТ”.

Результатом проведенного анализа бизнес-проблемы стало решение о создании СУТ на основе единого хранилища данных – репозитория (см. Приложение 2 “Причинно-следственный анализ”¹¹).

СУТ позволит представителям оперативного управления, исполнителям работ и участникам проекта «АСКО» разрабатывать требования и управлять ими в режиме реального времени, используя Интернет.

Разработать план управления требованиями (УТ) для системы АСКО, описывающий процесс УТ.

- Проработать введение плана.

Эта часть документа включает в себя цель плана УТ, краткое описание области применения документа, определения, сокращения и аббревиатуры, используемые в тексте плана, ссылки на используемые информационные источники.

- Определить Ответственных за выполнение задач УТ.
- Определить Вычислительные и программные средства, используемые при УТ, процедуры контроля версий требований.
- Составить Программу управления требованиями.
- Идентификация требований, отслеживание, документирование, управление изменениями, основные мероприятия и задачи.
- Назначить Контрольные точки.
- Определить, как будет проводиться подготовка кадров и ресурсов.

Разработать краткий регламент работы с системой УТ на RequisitePro.

Практические:

Создать репозиторий в RequisitePro для проекта АСКО.

- Установить ПО IBM Rational RequisitePro 7.1.
- Создать в репозитории Проект АСКО.
- Ввести и разместить все документы с требованиями для АСКО.
- Ввести глоссарий проекта как документ.
- Бизнес-требования к системе.
- Функциональные требования к системе.
- Создать сценарии использования инструмента УТ.

Продemonстрировать выполнение на RequisitePro основных задач работы с требованиями с акцентом на разработку форм ввода.

3.1 Постановка задачи на проектирование СУТ.

В выпускной квалификационной работе (ВКР) планируется разработать систему управления требованиями (СУТ) в Проекте “АСКО”. СУТ с функцией удалённого доступа должна автоматизировать разработку, управление требованиями, обеспечить единое хранение требований и проектной документации.

Новая СУТ должна позволять Менеджеру Проекта и Руководителю групп Исполнителей работ обсуждать и согласовывать, уточнять требования к АСКО, обеспечивать подготовку требований, в рамках его компетенции, запросы на изменения и проектные документы, координировать работы по Проекту, просматривать Отчёты о статусах требований и работ по Проекту, контролировать сроки исполнения требований Проекта в рамках Оперативного совета. (Оперативный совет должен проводиться не реже 1 раза в неделю под руководством Менеджера проекта от Функционального заказчика ([Устав АСКО.](#))).

СУТ должна позволять Исполнителям работ в Проекте создавать требования для:

- разработки ТЗ, ЧТЗ, проектной и рабочей документации,
 - проектирования и настройки АСКО,
- проводить рассмотрение и проверку документов, выдачу предложений и замечаний.

СУТ должна предоставить Проектному офису:

- Портал Проекта,
- ведение текущей организационно-распорядительной документации по проекту и архива проектной документации,
- ведение журнала модификаций и сводного реестра разработок,
- организацию процессов согласования и утверждения проектной документации,
- контроль статуса и оформление проектной документации,
- инфраструктурное обеспечение обучения пользователей и проведения тестирования Системы,
- возможность запрашивать у Менеджеров Проекта, руководителей проектных групп и подгрупп информацию и материалы, необходимые для эффективной работы Проектного офиса,
- внесение Оперативному совету предложения по оптимизации деятельности Проектного офиса,
- проведение Мониторинга изменений и актуализацию данных о персональном составе Участников Проекта,
- размещение данных о персональном составе Участников Проекта на Портале Проекта,
- разграничение прав доступа к СУТ.

Работы по Проекту выполняются совместно проектными группами со стороны Заказчика и Исполнителя. Проектный офис еженедельно (каждый понедельник) актуализирует информацию о состоянии Проекта (исполнение утвержденного графика работ, планов работ, протоколы заседаний

Управляющего комитета, оперативного совета и их исполнение и т.д.).
Электронные копии документов размещаются на Портале Проекта.

Для организации хранения документации по Проекту выделено дисковое пространство файловых серверов. Взаимодействие Проектных групп затруднено, т.к. выделены разобщённые рабочие подсети. Проектная группа в ходе работ по Проекту для ведения внутренней и отчетной документации использует следующий программный инструментарий:

- Microsoft Office 2003;
- Microsoft Project 2007.

Для обмена текущей информацией по Проекту в качестве основного канала используется электронная почта. Почтовый ящик проекта: asco@gazauto.gazprom.ru Почтовый ящик предназначен для использования в интересах Проектного офиса, в частности получения на него информационных материалов от участников проекта. Пользоваться этим почтовым ящиком в настоящий момент может только руководитель Проектного офиса, кроме того руководитель Проектного офиса администрирует Портал проекта. Портал доступен через сеть Интернет для зарегистрированных пользователей. Адрес Портала Проекта: <http://172.16.1.2:8080/default.aspx>

Требования на протяжении всего Проекта должны проходить процедуру согласования, которая включает в себя четыре стадии: стадию **подготовки**, стадию **проработки**, стадию **согласования**, стадию **утверждения**.

Для разных типов требований, определенных в Концепции, применяются разные процедуры согласования и утверждения.

Для организации работ с требованиями определены следующие роли:

- Разработчик – роль, отвечающая за подготовку, проработку требования и устранение замечаний на стадиях проработки и согласования.
- Рецензент – роль, отвечающая за формирование замечаний к требованию.

- Ответственный за согласование – роль, отвечающая за организацию процесса согласования методологических требований в подразделениях ОАО «Газпром».

- Участник согласования – роль, отвечающая за согласование технологических и эксплуатационных требований.

На стадии подготовки Разработчик требования подготавливает требование и доводит его до состояния, позволяющего содержательно обсуждать требование с участниками Проектной группы. На всех стадиях, которые проходит требование, Разработчик требования вносит в него изменения, устраняет или мотивирует необоснованность замечания. ФИО Разработчика требования фиксируется в Детальном плане работ.

На стадии проработки работу над требованием совместно ведут участники рабочих групп (Рецензенты) и Разработчик требования. Рецензенты фиксируют замечания к требованию и направляют их Разработчику требования. Разработчик требования устраняет или разумно отклоняет замечания. По запросу Разработчика требования Рецензенты фиксируют замечания в исходном требовании (файле) в режиме исправления или фиксируют замечания в xls-файле. Цикл обработки замечаний может быть повторен несколько раз в течение стадии проработки, продолжительность которой определена в Детальном плане работ.

По своему усмотрению Разработчик требования может закончить стадию проработки раньше указанного срока и отправить требование на согласование, предварительно получив одобрение Менеджера Проекта от Исполнителя. В настоящий момент Разработчик требования передает одобренный документ с требованиями через Проектный офис по электронной почте администратору Проектного офиса на согласование.

Согласование документа проходит по разным схемам для группы методологических документов и для группы организационных, технологических и эксплуатационных документов.

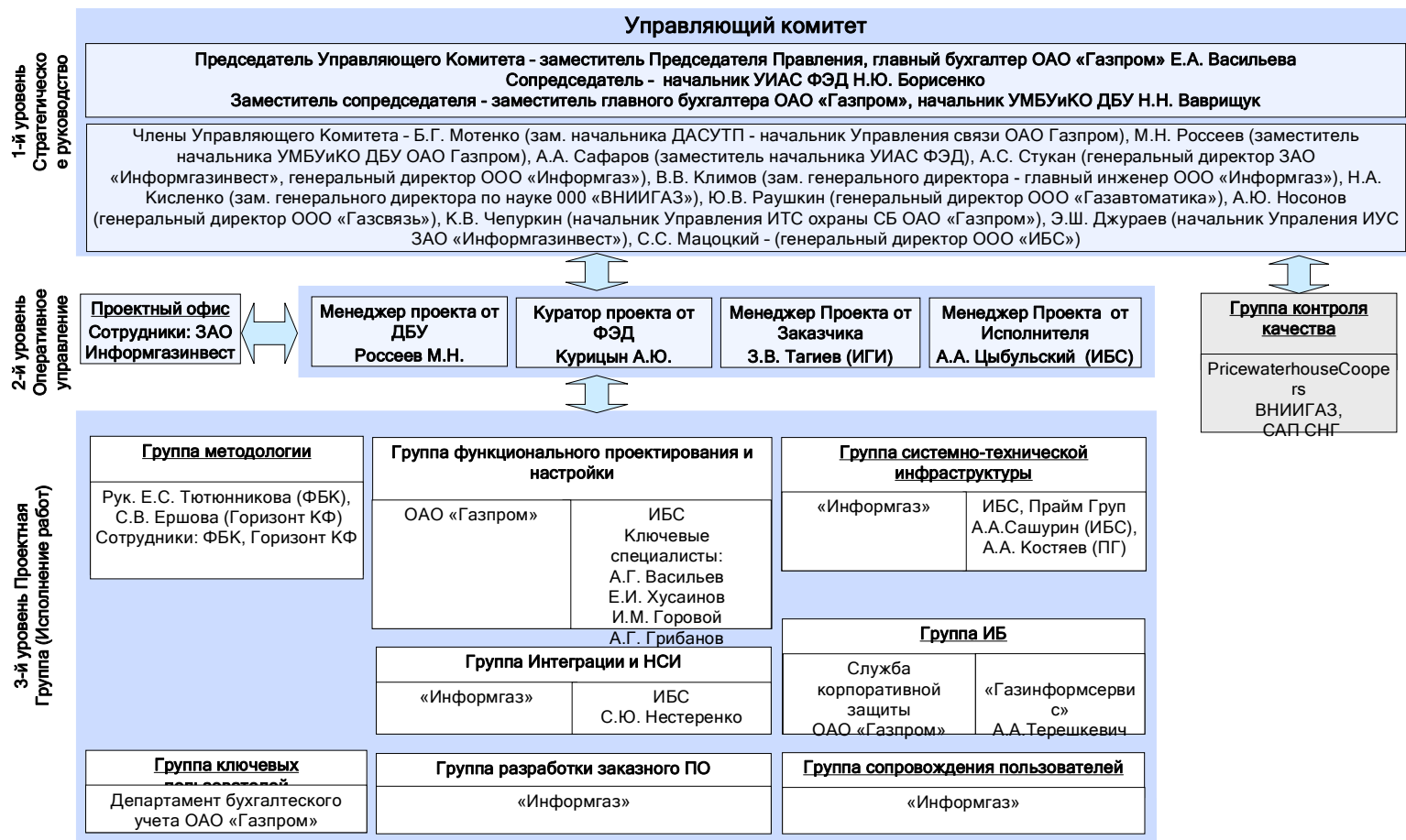


Рисунок 3-1 Организационная структура Проекта АСКО

Суммарно постановка задачи представлена ниже (Таблица 2):

Таблица 2

Проблема в:	Отсутствует регулярное управление требованиями.
Влияет на:	Представителей оперативного управления, исполнителей работ
Влияние заключается в:	Ошибки в требованиях самые распространенные и самые дорогие для исправления
Требования на успешное решение:	Автоматизация процесса управления требованиями. Единое хранение требований. Участие всех групп заинтересованных лиц в процессах разработки требований и управления ими в режиме он-лайн. Доступ к непротиворечивым, актуальным требованиям.

Результат постановки задачи:

Таблица 3

Целевые пользователи	Менеджеры, Исполнители работ
что	Посредством он-лайн доступа виртуально объединит Проектный офис
Существующий продукт	Репозиторий в IBM RequisitePro
Который	Снизит расходы Проекта за счёт автоматизации процесса управления требованиями
В отличие от других решений	MS Office (Word, Outlook), телефония Telelogic DOORS Analyst Real System 7.0 (ARTS)
Ключевые особенности продукта	Возможность создания регламента УТ, используя RUP Возможность интеграции с инструментами версионного контроля Формирование структуры требований (оговаривается в Плане УТ) Фиксирование исходных и возникающих требований (запросы заинтересованных лиц, переформулируемых в свойства, а потом в требования)

	Визуальное оформление требований (Rational Rose) Контроль статусов требований Определение необходимых типов требований и атрибутов для них (в Плане УТ) Возможность генерации актуального технического задания (SoDA) Учет истории и авторства изменений требований Анализы влияния изменений (change impact) () Сбор, обработка и использование статистики требований () Средства ведения форумов и дискуссий по требованиям Удалённая работа с требованиями
--	---

Общее число Участников Проекта, использующее СУТ = 130 человек.

Персональный состав Участников Проекта в течение срока реализации Проекта может меняться.

3.2 Определение процесса управления требованиями

Прежде чем составлять описание требований к проекту, необходимо выбрать способ их оформления и систематизации, а также способ применения атрибутов требований при управлении требованиями в процессе работы над проектом.

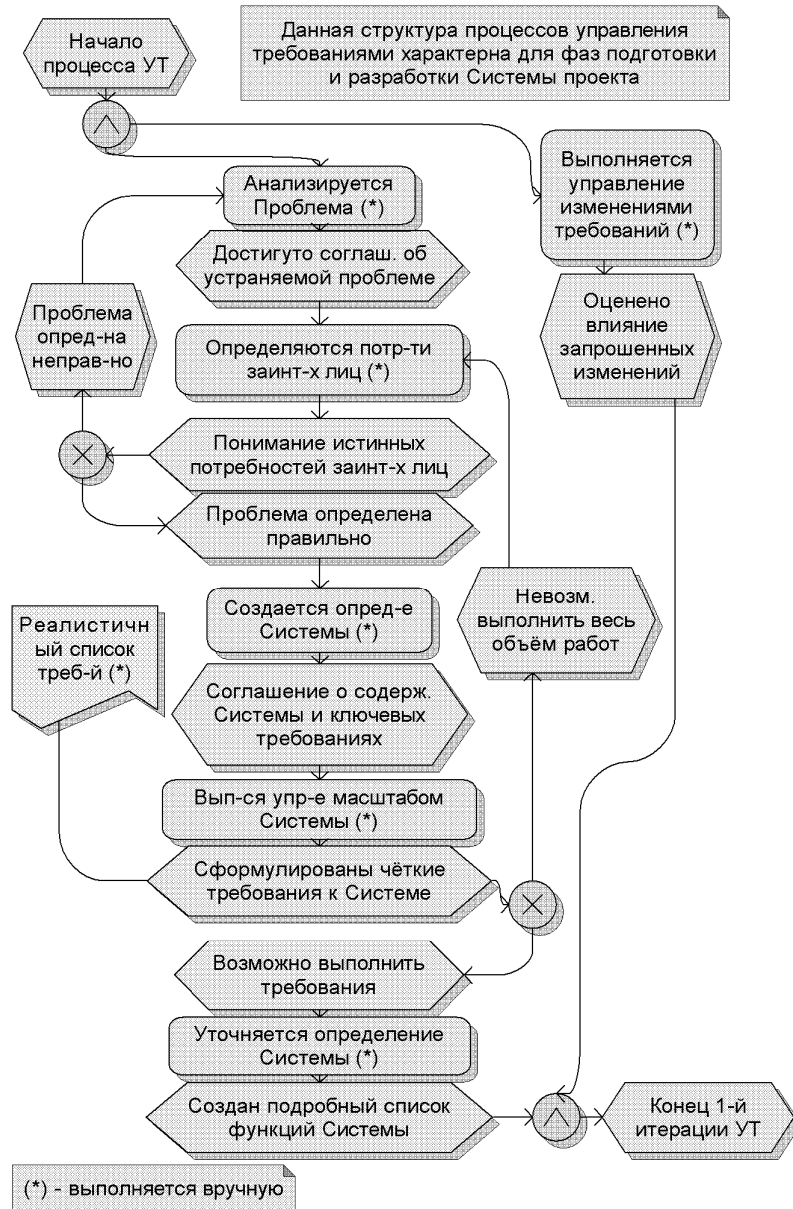
Выбор атрибутов поможет в следующем:

- Оценить, как изменения требований влияют на проект.
- Оценить влияние неудачного тестирования, основанного на требованиях тестирования.
- Управлять рамками проекта.
- Убедиться, что все требования к системе выполнены к началу реализации.
- Убедиться, что приложение выполняет только те функции, которые оно должно выполнять.
- Управлять изменениями.

Все решения, касающиеся документов с требованиями, элементов трассируемости, рекомендаций и стратегий для работы с атрибутами требованиями вошли в план управления требованиями.

4 План управления требованиями определяет взаимоотношения и заинтересованных лиц и их ответственность, отслеживаемые элементы, управление изменением требований и отчеты по выполнению требований (см. Приложение 3 “Процесс управления требованиями в проекте АСКО”).

Бизнес-процесс "Управление Требованиями"



Правильная связь с заинтересованными сторонами может помочь с ошибками, связанными с требованиями. Заинтересованные стороны играют важную роль в определении требований и понимании их приоритетов. Большинство заинтересованных лиц, однако, плохо разбираются во влиянии предъявляемых требований на стоимость и график и, следовательно, организация, занимающаяся разработкой, должна самостоятельно работать с ожиданиями заказчиков.

Установка отношений с заинтересованными сторонами включает определение:

- Ответственности заинтересованных лиц: будет ли заказчик доступен для консультаций? В заранее обговоренное время?
- Видимость рабочих продуктов заказчику: все ли рабочие продукты должны предоставляться? Предоставлять только запланированные вехи?

Для эффективного управления необходимо выделять различные уровни и цели требований, для этого вводится понятие типов требований. Т.к. проект объёмный и сложный формулировки бизнес-правил и концепции ведут к потребностям пользователя, функциям и требованиями к АСКО, программные требования – к требованиям разработки, тестирования, документации. Типы требований – это типичные элементы трассируемости. С каждым элементом трассируемости связан собственный набор атрибутов, который применяется для анализа состояния, преимуществ, рисков и прочих факторов, связанных с элементом.



Рисунок 4-1 Структура трассируемости

Трассируемость (Прослеживаемость) позволяет выполнить следующее:

- Понять источник требований.
- Обозначить область проекта.
- Работать с изменениями требований.
- Оценить воздействие изменений в требованиях на проект.
- Оценить воздействие сбоя теста на требования (например, сбой

теста может означать, что требования не выполнены).

- Убедиться, что реализация системы отвечает всем требованиям.
- Убедиться, что приложение делает только то, что от него

требуется.

Трассируемость (Прослеживаемость) помогает понять, как входные требования, такие как бизнес-правила и запросы заинтересованных лиц, преобразуются в ключевые потребности заинтересованных лиц и пользователей и системные функции. Трассируемость (Прослеживаемость) также обеспечивает возможность контроля за тем, как эти требования воплощаются в проекте, как они тестируются, и как документируются для пользователя.

Ключевым в работе с изменениями требований является понятие "недостовойной" связи трассируемости. Когда изменяется требование или другой элемент трассируемости, все связи трассируемости, ведущие к этому элементу, помечаются как недостовйные. Ответственные за эти связи должны будут изучить изменение и определить, требуется ли изменение также и соответствующих элементов. Это понятие также способствует анализу влияния потенциальных изменений.

Элементы трассируемости также позволяют ответить на следующие вопросы:

- Какие потребности пользователя не связаны с возможностями продукта.
- Каково состояние тестов для всех вариантов использования в итерации #n.
- Каковы вспомогательные требования для тестов, статус которых не проверен.
- Какие тесты не были пройдены, в порядке важности.
- Какие возможности запланированы для данного выпуска, какие запросы пользователей они выполняют, и каково их состояние.

Атрибуты требований позволяют отслеживать информацию, связанную с элементами трассируемости. Обычно эти данные используются для определения статуса и составления отчетов. Отслеживать необходимо атрибуты риск, польза, трудозатраты, стабильность и влияние на архитектуру, чтобы можно было распределять требования между итерациями и классифицировать по важности для управления рамками проекта.

Необходимость изменений неотъемлема от разработки системы программного обеспечения, поскольку система развивается при первоначальном ее создании, и в последующем применяется и обслуживается в процессе ежедневной работы в реальной среде. Запросы на изменения предоставляют запись решения и, с помощью соответствующего процесса оценки, гарантируют, что влияние изменений является обдуманнм.

Запросы на изменение также известны как CR, дефекты, ошибки, инциденты, запросы на расширения. Соответственно фиксация этих запросов и управление ими гарантирует, что изменения вносятся в систему под контролем, так что их влияние на систему предсказуемо. Ниже перечислены некоторые важнейшие типы запросов на изменение.

Запросы на расширение применяются различными заинтересованными лицами для запроса будущих функциональных возможностей, которые они хотели бы включить в продукт. Этот вид запроса трассируется с Запросом заинтересованных лиц, который фиксирует и формулирует понимание потребностей заинтересованных лиц.

Дефекты представляют собой отчеты об аномалиях или сбоях в конечном рабочем продукте. Дефекты включают пропуски и неполноту, обнаруженные на начальных этапах жизненного цикла, либо признаки сбоев, которые необходимо изолировать и скорректировать в программном продукте. Дефекты также могут включать отклонения от ожидаемого алгоритма работы программы (например, вопросы, связанные с удобством работы).

Цель дефекта - сообщить сведения о неполадке, вызвав корректирующее действие, разрешение вопроса, и проследив происхождение неполадки.

Следующий пример диаграммы показывает типовые состояния и указывает, кого необходимо уведомлять в течение жизненного цикла запроса на изменение:

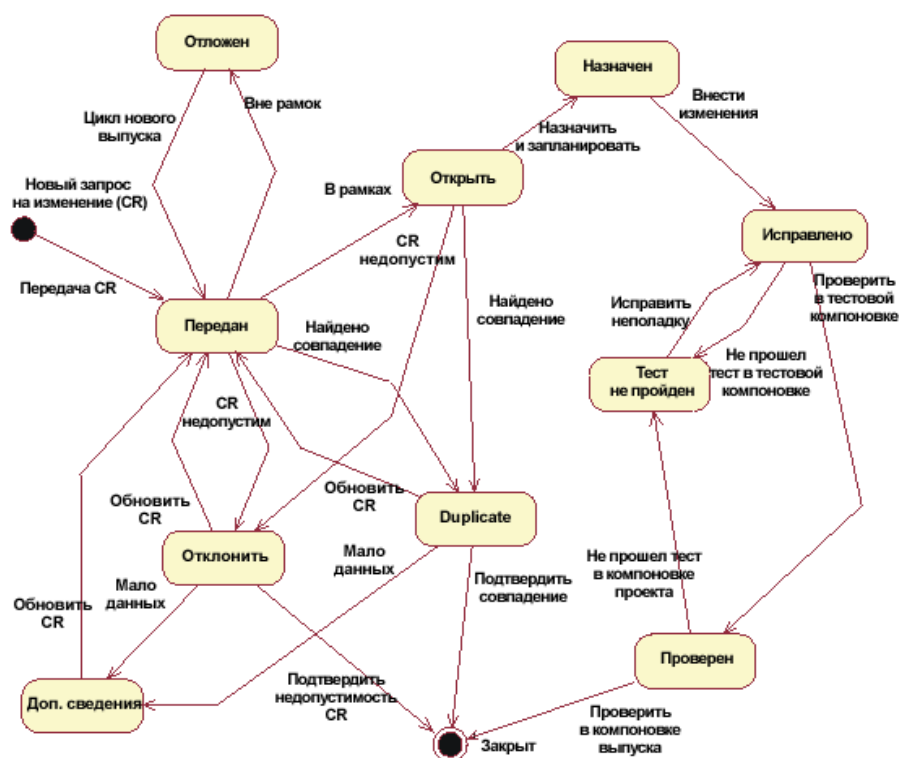


Рисунок 4-2 Диаграмма состояний запроса на изменение

Описание примера состояний процесса управления запросами на изменение:

Таблица 4 Описание состояний запроса на изменение

Состояние	Определение	Права доступа
Подан	Это состояние возникает в следующих ситуациях: 1) подачи нового запроса изменений, 2) обновления существующего запроса на изменение или 3) рассмотрения отложенного запроса на изменение для следующего выпуска. Запрос изменений помещен в очередь для рассмотрения Советом по контролю за изменениями. На этом этапе ему не присваивается ответственное лицо.	Все пользователи
Отложен	Запрос изменений признан допустимым, но вне рамок данного выпуска. Запросы изменений со статусом Отложен рассматриваются при	Администратор Руководитель проекта

Состояние	Определение	Права доступа
	последующих выпусках продукта. Также для запроса на изменение может быть указан целевой выпуск и его время выхода, для которого запрос считается поданным и вновь помещается в очередь для рассмотрения Советом по контролю за изменениями.	
Повторный	Запрос изменений считается повтором поданного запроса на изменение. Присвоить этот статус запросу может Администратор Совета по контролю за изменениями или ответственное лицо. При присвоении запросу статуса Повторный будет указан номер запроса, который он повторяет (в ClearQuest - на вкладке Вложения). Подающий запрос должен запрашивать базу данных запросов перед подачей нового запроса. Это позволит сэкономить время. Подавшие повторный запрос изменений должны быть занесены в список для оповещения о состояниях этого запроса на изменение.	Администратор Руководитель проекта Администратор контроля качества Разработка
Отказ	Этот статус запроса на изменение означает, что он признан недопустимым Советом по контролю за изменениями или уполномоченным участником команды или требуется дополнительная информация. Если ранее запрос изменений обладал статусом Открыт , то он убирается из очереди на выполнение и будет вновь рассмотрен. Для подтверждения этого назначается участник Совета по контролю за изменениями. От подавшего запрос не требуется дополнительных действий, за исключением случая, когда состояние запроса изменяется на Дополнительная информация . Запрос изменений будет рассмотрен Советом по контролю за изменениями с учетом дополнительной	Администратор Руководитель проекта Администратор разработки Руководитель тестирования

Состояние	Определение	Права доступа
	информации. В случае признания недопустимым, запрос будет закрыт, и подавший запрос будет уведомлен.	
Дополнительная информация	Для проверки допустимости запроса на изменение со статусом Отказ или Повтор недостаточно данных. Подавший запрос уведомляется о недостаточности данных.	Администратор
Открыт	Запрос изменений был признан допустимым в рамках данного выпуска и ожидает выполнения. Его выполнение назначено на определенную веху проекта. Это называется - быть в "очереди назначения". Только члены Совета по контролю за изменениями могут перевести запрос в очередь на выполнение. Если обнаружен запрос изменений с приоритетом два или выше, он должен быть немедленно направлен на рассмотрение администратору разработки или Администратору контроля качества. Они могут запланировать внеочередную встречу Совета по контролю за изменениями или просто поместить запрос в очередь на выполнение.	Администратор Руководитель проекта Администратор разработки Отдел контроля качества
Назначен	В обязанности руководителя проекта входит назначить ответственного за открытый запрос изменений, в зависимости от типа запроса на изменение, и обновить расписание проекта.	Руководитель проекта
Выполнен	Означает, что запрос выполнен и нуждается в проверке. Если подавший запрос - участник отдела контроля качества, он автоматически становится владельцем запроса, в противном случае владельцем становится администратор контроля качества и запрос назначается вручную.	Администратор Руководитель проекта Администратор разработки Администратор контроля качества

Состояние	Определение	Права доступа
		Отдел разработки
Тест не пройден	Этот статус присваивается запросу, который не прошел тестирование в тестовой или выпускной компиляции. Владелец запроса становится участник команды, выполнивший изменения.	Администратор Отдел контроля качества
Проверен	Этот статус присваивается запросу, проверенному в тестовой компиляции, и готовому к включению в выпуск.	Администратор Отдел контроля качества
Закрыт	Запрос изменений не требует дальнейшей обработки. Это последний из возможных статусов запроса на изменение. Этот статус может присваивать только администратор проверки Совета по контролю за изменениями. При присвоении статуса Закрыт подавший запрос получает уведомление. Статус Закрыт может быть присвоен в следующих случаях: 1) после проверенного выполнения в выпускной компиляции, 2) после подтверждения статуса Отказ или 3) при обнаружении повтора другого запроса. В последнем случае подавший запрос будет уведомлен о наличии такого запроса и добавлен к списку уведомляемых о статусе первоначального запроса (за дополнительной информацией обратитесь к определениям статусов Отказ и Повтор). Если подавший запрос хочет оспорить закрытие, то он может обновить запрос и вновь подать его на проверку Совета по контролю за изменениями.	Администратор

Теги статуса позволяют создавать статистические отчеты о запросах на изменение (возраст, распределение, тенденции).

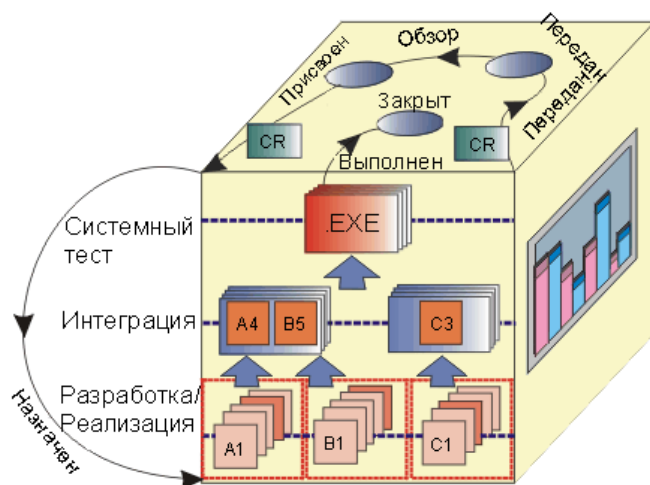


Рисунок 4-3 Состояние запросов на изменение в контексте куба

Выбор отчетов зависит от доступных инструментов составления отчетов. При наличии таких инструментов рекомендуется создавать отчеты для рабочих продуктов, основанных на моделях или базах данных.

4.1 Шаблоны документов

Состав и комплектность проектной документации определяется Исполнителем и согласовывается с Заказчиком и Функциональным заказчиком АСКО. Проектная документация АСКО не должна противоречить руководящим и нормативным документам федерального уровня и требованиям документов ОАО «Газпром».

Состав проектной документации, выпускаемой по результатам проведения работ на различных фазах, приведен ниже (Таблица 5Таблица 6).

Таблица 5

№	Наименование фазы работ	Состав документации
1	Подготовка проекта	Устав проекта Техническое задание на создание АСКО Частное техническое задание на разработку ПОИБ АСКО Календарный план Проекта Письменный отчет с замечаниями и предложениями к

№	Наименование фазы работ	Состав документации
		техническому заданию
2	Разработка Рабочего проекта	Методология консолидации Регламент формирования Консолидированной отчетности Рабочий проект Требования к бизнес-процессу «Бухгалтерский учет» Шаблона ИУС П Комплект документации на КТС АСКО Уточненный Календарный план Проекта на фазу 3 Письменный отчет с замечаниями и предложениями к методологии консолидации, бизнес-проекту, перечню данных в АСКО
3	Разработка системы	Комплект документов по вводу в действие КТС АСКО Проектные решения Рабочая документация на ПОИБ Спецификации на разработки Технологические инструкции Операционные инструкции Матрица доступа пользователей к информационным и функциональным ресурсам План и программа обучения ключевых, конечных пользователей, группы по информационной безопасности, эксплуатационного персонала Документы к аттестационным испытаниям Ведомость обучения подгруппы информационной безопасности Протокол обучения ключевых пользователей Сценарии предварительных испытаний Регламент предварительных испытаний Программа и методика испытаний ПОИБ Результаты (протокол) предварительных испытаний План ввода системы в Опытную эксплуатацию Регламент подключения пользователей к АСКО Письменный отчет с замечаниями и предложениями по итогам

№	Наименование фазы работ	Состав документации
		проведения предварительных испытаний системы
4	Подготовка системы к опытной эксплуатации	Решение УК о начале Опытной эксплуатации системы Протоколы обучения конечных пользователей Технологические и операционные инструкции Письменный отчет с замечаниями и предложениями к анализу переноса входящих остатков
5	Опытная эксплуатация системы	Протокол по результатам опытной эксплуатации Описание настроек и разработок системы Письменный отчет с замечаниями по опытной эксплуатации системы
6	Передача системы в опытную эксплуатацию	Описание настроек и разработок системы (с доработками и уточнениями) Технологические инструкции (с доработками и уточнениями) Операционные инструкции (с доработками и уточнениями) Программа и методики испытаний Программа и методики испытаний ПОИБ Решение о создании комиссии по приемке системы в постоянную эксплуатацию Протокол приемки документации по настройкам и разработкам Протокол приемки системы в постоянную эксплуатацию (протокол испытаний) Письменный отчет с замечаниями по результатам ввода системы в постоянную эксплуатацию

В процессе создания АСКО, состав документации может уточняться по согласованию между Заказчиком и Исполнителем.

Документация выпускается на бумажных носителях и электронных носителях.

Каждый новый документ в RequisitePro создается исходя из типа документа. Тип документа включает в себя расширение файла по умолчанию, тип требований по умолчанию, текст и форматирование по умолчанию (касающиеся шрифта, табуляций и т.д.) и контролируется

связанным с ним шаблоном. RequisitePro предоставляет следующие схемы для разработки требований и прецедентов:

- Документ с требованиями к продукту;
- Спецификации требований программного обеспечения;
- Современные спецификации требований программного обеспечения;
- Спецификация нескольких прецедентов;
- Документ с тестовыми требованиями;
- Прецеденты функциональных тестов;
- План управления требованиями;
- Запросы заинтересованных лиц;
- Концепция;
- Глоссарий;
- Спецификации прецедентов;
- Спецификации требований программного обеспечения (с вариантами);
- Спецификации требований программного обеспечения (без вариантов);
- Дополнительная спецификация;
- План тестирования.

Можно выделить 5 типов документов, использующихся Проектным офисом:

- Концептуальный подход (К);
- Методология (М);
- Техническое задание (ТЗ);
- Проектное решение (ПР);
- Аналитическая записка (АЗ).

С каждым из этих типов документов будут связаны свои шаблоны документов. См. рис.

4.2 Инструментальная поддержка процесса управления требованиями.

Приложение IBM® Rational® RequisitePro® помогает коллективам организовывать, назначать приоритеты, отслеживать и контролировать изменяющиеся требования к системе или приложению.

Rational RequisitePro, как и все другие инструменты Rational, поддерживается международной обслуживающей организацией, которая предлагает услуги по обучению, консультации и технической поддержке, помогающие клиентам Rational ускорить внедрение технологии и выполнение проектов.

Rational RequisitePro предназначена для улучшения обмена информацией, повышения эффективности сотрудничества в группе и снижения риска проекта. Ключевой особенностью RequisitePro является его тесная интеграция с Microsoft Word – несомненно, наиболее широко используемым инструментом для создания документов и электронного обмена текстовой информацией. Поскольку предложение, обновление, проверка и изменение требований достаточно часто выполняются различными техническими и деловыми специалистами с помощью документов Word, эта интеграция позволяет проектным группам продолжить использование знакомого инструмента.

Возможности Rational RequisitePro позволяют организовать требования, установить их приоритет, отследить взаимосвязи и изменения требований, одновременно поддерживая динамическую связь с исходным документом Word.

Хотя документы удобны для фиксирования требований, они не позволяют эффективно их организовать и определить приоритеты. Возможности Rational RequisitePro позволяют организовать требования, установить их приоритет, отследить взаимосвязи и изменения требований,

одновременно поддерживая динамическую связь с исходным документом Word.

Пакет Rational Suite AnalystStudio предназначен для задач управления требованиями и их анализа при разработке приложений. Он включает в себя инструменты и процессы Rational, поддерживающие управление требованиями в течение всего жизненного цикла разработки ПО. Ключевыми компонентами Rational Suite AnalystStudio являются Rational RequisitePro (управление требованиями) и Rational Rose (моделирование бизнеса, прецедентов, данных и приложений). Интеграция Rational RequisitePro и Rational Rose позволяет пользователям зафиксировать в RequisitePro требования в текстовом виде (в виде прецедентов, описывающих представление требований конечным пользователем, либо в другой форме) и одновременно создать соответствующую диаграмму прецедентов в Rational Rose. Изменения, вносимые в одном из инструментов, отображаются и в другом, что позволяет любому члену группы получить единую информацию о требованиях из любого инструмента.

Варианты использования Rational RequisitePro в проекте АСКО представлены в приложении (см. Приложение 4 “Сценарии использования СУТ”).)

4.3 Выводы

Основным результатом Главы 2 стала постановка задачи на проектирование новой СУТ на базе Rational RequisitePro. Так же были сформулированы верхнеуровневые требования к СУТ: система должна позволять Менеджеру Проекта и Руководителю групп Исполнителей работ обсуждать и согласовывать, уточнять требования к АСКО, обеспечивать подготовку требований в рамках его компетенции, запросов на изменения и проектные документы, координировать работы по Проекту, просматривать Отчёты о статусах требований и работ по Проекту, контролировать сроки исполнения требований Проекта. Разработана методика, представленная в

плане управления требованиями для системы АСКО, описывающая процесс УТ, проанализирована структура Проектного Офиса и выделены документы Проекта АСКО, шаблоны которых будут использоваться при управлении требованиями. Представлены EPC и UML – диаграммы, наглядно демонстрирующие бизнес-процесс УТ, как есть (“AS IS”) и как должен стать после автоматизации (“TO BE”). Создан репозитория в RequisitePro для проекта АСКО, хранящий документы с требованиями к АСКО, глоссарий проекта, бизнес-требования к системе, функциональные требования, сценарии использования инструмента УТ, план УТ.

Апробация и тиражирование результатов возможно при внедрении решения на этапе сопровождения проекта АСКО.

5 Глава 3 “Управление требованиями на этапе сопровождения АСКО”.

5.1 Процесс управления изменениями в терминах ITIL.

Быстрое развитие ИТ-технологий и рынка привели к тому, что сейчас изменения стали обычным делом. Однако опыт показывает, что инциденты, влияющие на бизнес-приложения, часто бывают вызваны изменениями. Причины таких инцидентов могут быть различными: халатность сотрудников, недостаток ресурсов, недостаточная подготовка, слабый анализ воздействия изменения, несовершенство испытаний или «болезни роста». Если инциденты, связанные с изменениями, не будут контролироваться, из-под контроля может выйти все предоставление ИТ-услуг и сам бизнес. Число инцидентов может увеличиваться, каждый из них будет требовать принятия срочных мер, что в свою очередь может привести к возникновению новых инцидентов. Ежедневное планирование часто не в состоянии учитывать увеличивающуюся рабочую нагрузку. Это может повлиять на повседневную работу и на сопровождение ИТ-услуг.

Целью Процесса Управления Изменениями является руководство проведением изменений и ограничение числа инцидентов, вызванных изменениями. Девиз Процесса Управления Изменениями: “Не всякое, изменение является улучшением, но всякое ухудшение является изменением”.

На Рисунок 5-1 показан цикл изменений, вызванный предложениями на новые разработки и улучшения (Процессы Предоставления услуг и Управления Проблемами), Запросами (адресованные в Процесс Управления Изменениями) и требуемыми решениями (Процесс Управление Проблемами):

- **Инновации и усовершенствования** - внедрение новых услуг и новых технических средств в ИТ инфраструктуру становится причиной

появления новых регулярно возникающих ошибок (Long-term errors) в ИТ-инфраструктуре.

- **Изменения** - все, от небольших инсталляций до перестановки мейнфреймов, при неаккуратном проведении изменений могут стать причиной появления новых регулярно возникающих ошибок в ИТ-инфраструктуре.

- **Корректирующие меры** - нацелены на исправление недавно появившихся регулярно возникающих ошибок.

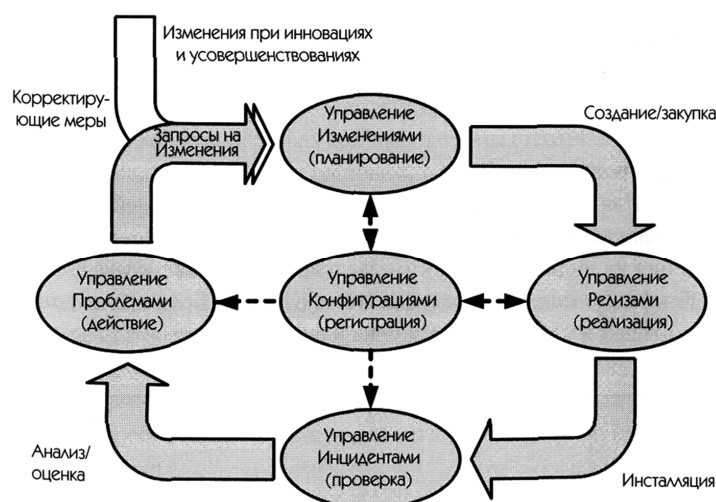


Рисунок 5-1 Входы Процесса Управления Изменениями

Управление Изменениями действует как терморегулятор между гибкостью (одобрение изменений, которые могут привести к долгосрочным ошибкам) и стабильностью (одобрение изменений для исправления долгосрочных ошибок). Корректирующие меры уменьшают число инцидентов, за счет чего снижается рабочая нагрузка. Возвращаясь к аналогии с терморегулятором, можно сказать, что температура воды соответствует Уровню Изменений и Инноваций, с которой организация может справиться.

5.2 Основные термины

Две точки принятия решения об изменениях.

Существуют две точки принятия решения о проведении изменений:

1. **Руководитель Процесса Управления Изменениями** - лицо, ответственное за предварительный просмотр (фильтрацию) и классификацию Запросов на Изменения (Request For Change - RFC). В больших организациях в помощь Руководителю Процесса могут существовать Координаторы изменений, осуществляющие взаимодействие между ним и различными подразделениями организации. Одной из задач Процесса Управления Изменениями является получение требуемой авторизации изменения. В определенной степени сам процесс уже имеет полномочия, но для проведения некоторых изменений может потребоваться согласование с руководством ИТ (например, с Руководящим комитетом (Steering Committee) или Исполнительным комитетом (Executive Committee)). Руководитель Процесса Управления Изменениями также несет ответственность за планирование и координацию проведения изменений.

2. **Консультативный комитет по изменениям** (Change Advisory Board - CAB) - консультативный орган, регулярно собирающийся для оценки и планирования изменений. Чаще всего одно или несколько значительных изменений выносятся на обсуждение Комитета. Отдельно создается комитет по срочным изменениям (Emergency Committee - EC), который назначается руководством для принятия чрезвычайных решений. Состав Консультативного комитета является гибким и включает представителей всех основных ИТ-подразделений:

- Руководителя по Управлению Изменениями (председатель);
- Руководителя по Управлению (Уровнями) Услуг;
- Представителей Службы Service Desk и Процесса Управления Проблемами;
- Руководителей линейных подразделений компании;
- Бизнес-руководителей (или их представители) со стороны заказчика;
- Представителей пользователей;

- Представителей разработчиков приложений; Администраторов программного обеспечения и системных администраторов;
- Представителей поставщиков.

Охват (сфера действия или границы, Scope) процесса.

Сфера действия Процесса Управления Изменениями определяется вместе со сферой действия Процесса Управления Конфигурациями, так как Управление Конфигурациями предоставляет информацию для оценки воздействия изменения на инфраструктуру. После проведения изменения Управление Конфигурациями обновляет Конфигурационную Базу Данных (CMDB). Если в базе данных CMDB ведется учет компьютерных мышей и клавиатур, то замена клавиатуры рассматривается как изменение. Определение сферы действия процесса является динамической деятельностью, так как сфера действия может меняться, и потребность в информации из базы данных CMDB также меняется. Следовательно, сфера действия должна регулярно пересматриваться, и модель данных CMDB должна обновляться соответственно.

Для того, чтобы обеспечить эффективное взаимодействие Процессов Управления Изменениями и Управления Конфигурациями, необходимо регистрировать изменения и обновлять соответствующие записи в CMDB. Можно допустить, что ряд повседневных заданий, точно определенных и подчиняющихся установленным процедурам (т. е. стандартных заданий), не требуют контроля ее стороны Процесса Управления Изменениями. Примерами таких заданий могут быть установка кассет для резервного копирования, создание идентификаторов (ID) пользователей и т. д. Эти действия не обрабатываются как изменения, а самое большее классифицируются как Запросы НЕ Обслуживание в рамках Процесса Управления Инцидентами. Внимательное изучение стандартных действий может быть полезно для предотвращения излишней бюрократизации Процесса Управления Изменениями.

Одним из способов выполнения таких действий является определение так называемых «предварительно авторизованных» (Pre-authorized) изменений (или «категории 0»), которые записываются в базу данных изменений (предпочтительно самими запрашивающими), но не требуют использования процедур по Управлению Изменениями. Например, если при приеме на работу нового сотрудника обычно выполняются четырнадцать действий (создание новой учетной записи (Account), настройка рабочей станции электронной почты и т. д.), то эти действия не требуют столь внимательного изучения, как значительные изменения инфраструктуры. Такой вид стандартных изменений обрабатывается по типовому шаблону как предварительно авторизованный «Запрос на Обслуживание».

5.3 Цель процесса.

Целью Процесса Управления Изменениями является гарантия использования стандартных методов и процедур для быстрой обработки изменений с минимальным возможным отрицательным воздействием изменения на качество услуг. Все изменения должны быть отслеживаемыми, чтобы можно было ответить на вопрос: «Что изменилось?»

5.3.1 Преимущества использования процесса.

Для эффективного предоставления ИТ-услуг организация должна уметь обрабатывать большое количество изменений с надлежащим Уровнем Ответственности при принятии решений.

Преимуществами Процесса Управления Изменениями являются:

- уменьшение отрицательного воздействия изменений на качество ИТ-услуг;
- более точные оценки затрат для предлагаемых изменений;
- уменьшение количества изменений, потребовавших возврата к исходному состоянию, и каждый такой возврат происходит более гладко;

- предоставление руководству более полной информации об изменениях, что позволяет выявлять проблемные области;
- повышение производительности работы пользователей за счет более высокой стабильности и качества ИТ-услуг;
- повышение производительности работы ИТ-персонала, не отрывающегося от плановой работы для проведения срочных изменений и процедур возврата;
- рост способности компании проводить частые изменения без нарушения стабильности ИТ-среды.

5.4 Процесс.

Процесс Управления Изменениями принимает или отклоняет каждый Запрос на Изменение (RFC). Руководитель Процесса Управления Изменениями содействует работе процесса, но реальные решения о наиболее значительных изменениях принимаются консультативным комитетом по изменениям (СAB). Членами комитета САВ являются представители разных отделов компании, а также заказчиков и поставщиков. Ответственность за предоставление информации о потенциальном воздействии предлагаемых изменений несет Процесс Управления Конфигурациями.



Рисунок 5-2 Позиционирование Процесса Управления Изменениями

Входы Процесса Управления Изменениями включают в себя:

- Запросы на Изменения (RFC);
- информация из базы данных CMDB (в частности, анализ степени воздействия изменений);
- информация из других процессов (из Базы данных мощностей CDB, информация о бюджете и т. д.);
- планирование изменений (Согласованный план изменений Forward Schedule of Change - FSC).

Выходы процесса включают:

- обновленный план изменений (Согласованный план изменений FSC);
- моменты инициирования действий (триггеры) в рамках Процессов Управления Конфигурациями и Управления Релизами;
- повестка дня Консультативного комитета CAB, протоколы и принятые решения;
- отчеты по Процессу Управления Изменениями.

Управление Изменениями имеет описанную ниже взаимосвязь с другими процессами.

5.4.1 Управление Инцидентами.

Процесс Управления Инцидентами имеет двухстороннюю связь с Процессом Управления Изменениями. С одной стороны, Управление Изменениями обрабатывает направляемые Управлением Инцидентами Запросы на Изменения для разрешения инцидента или запрашиваемые Управлением Проблемами изменения, устраняющие причину инцидента. С другой стороны, несмотря на многочисленные предосторожности, внедрение изменений все же может привести к возникновению инцидентов. Это может быть связано с ошибками проведения изменения или с недостаточной подготовкой пользователей к изменениям. Соответствующий персонал Управления Инцидентами должен быть информирован о проведении изменений, чтобы иметь возможность быстро определить и устранить возникающие инциденты.

5.4.2 Управление Конфигурациями.

Управление Изменениями и Управление Конфигурациями являются настолько тесно связанными процессами, что они могут быть эффективно интегрированы между собой - шаг, рекомендованный в библиотеке ITIL.

Изменения регистрируются под контролем Процесса Управления Конфигурациями, анализ воздействия изменений также проводится с участием Процесса Управления Конфигурациями. Управление Конфигурациями определяет зависимость между Конфигурационной Единицей CI (вовлеченной в проводимое изменение) и другими CI, чтобы определить, на какие другие элементы будет воздействовать это изменение.

5.4.3 Управление Проблемами.

Взаимосвязь между Процессами Управления Изменениями и Управления Проблемами во многом похожа на такую же связь между Процессами Управления Изменениями и Управления Инцидентами. С одной стороны, изменения часто бывают необходимы для разрешения проблем. С

другой стороны, если проведение изменений недостаточно контролируются, они могут привести к новым проблемам.

5.4.4 Управление Релизами.

Изменения часто приводят к необходимости разработки и распространения новых приложений или установке технической инфраструктуры. Это осуществляется с помощью Процесса Управления Релизами. Контроль над распространением новых версий осуществляется Процессом Управления Изменениями.

5.4.5 Управление Уровнем Сервиса.

Процесс Управления Уровнем Сервиса вовлечен в определение степени воздействия изменений на предоставление услуг и бизнес-процессы. В зависимости от ситуации в Консультативном комитете (CAB) могут участвовать представители Процесса Управления Уровнем Сервиса. Если изменение оказывает значительное воздействие или связано с высоким риском, его внедрение и сроки должны всегда обсуждаться с заказчиком. Управление Изменениями направляет в Управление Уровнем Услуг отчет «Проектируемая доступность услуг» (Projected Service Availability - PSA). В этом отчете Управление Изменениями излагает изменения в имеющихся Соглашениях об Уровне Услуг (SLA) и воздействие Согласованного плана изменений (FSC) на доступность услуг.

5.4.6 Управление Доступностью.

Процесс Управления Доступностью инициирует изменения, направленные на повышение доступности услуг и проверяет, привели ли предпринимаемые меры к ожидаемому результату. Управление Доступностью часто привлекается при оценке потенциального воздействия изменений, так как это воздействие может повлиять на доступность услуги.

5.4.7 Управление Мощностями.

Руководитель Процесса Управления Мощностями в первую очередь занимается вопросом анализа совокупного эффекта по результатам изменений в течение продолжительного периода времени, например, увеличением времени реакции приложений или потребностью в большей емкости для хранения информации. На основе составленного Плана мощностей Управление Мощностями регулярно предлагает усовершенствования и инициирует изменения в форме Запросов на Изменения (RFC).

5.4.8 Управление Непрерывностью ИТ-услуг.

Превентивные мероприятия и планы восстановления, гарантирующие непрерывность услуг, должны постоянно контролироваться, так как изменения инфраструктуры могут сделать эти планы неосуществимыми или избыточными. Процесс Управления Изменениями действует в тесной взаимосвязи с Процессом Управления Непрерывностью ИТ-услуг, чтобы в нем учитывались все изменения, которые могут повлиять на планы восстановления, и предусматривались меры, необходимые для проведения восстановления.

5.4.9 Виды деятельности в рамках Процесса Управления Изменениями.

Процесс Управления Изменениями включает в себя следующие виды деятельности для обработки изменений:

- Направление Запроса - не включается в виды деятельности по Управлению Изменениями, но поддерживается этим процессом, так как Управление Изменениями отвечает за правильную регистрацию всех изменений.
- Прием в обработку - предварительный просмотр (фильтрация) Запросов на Изменения и прием их к дальнейшему рассмотрению.

- Классификация - сортировка Запросов на Изменения по категориям и приоритету.
- Планирование - объединение изменений, планирование их проведения и планирование необходимых ресурсов.
- Координация - координирование компоновки, испытаний и проведения изменений.
- Оценка - оценка успешности каждого изменения и составление заключения для будущей деятельности (накопление знаний).

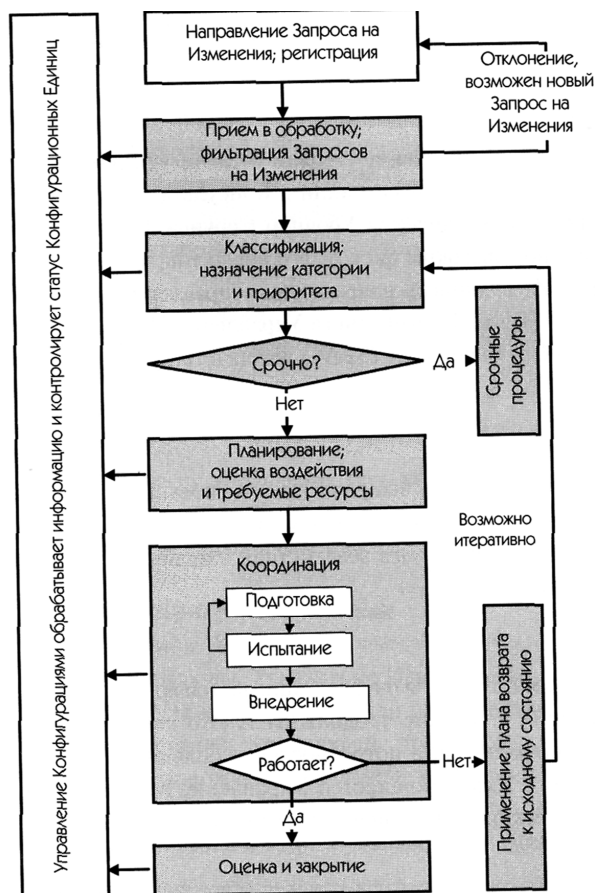


Рисунок 5-3 Виды деятельности в рамках Процесса Управления Изменениями

5.5 Виды деятельности в рамках Процесса Управления Изменениями.

5.5.1 Регистрация.

Прежде всего, все Запросы на Изменения (RFC) должны быть зарегистрированы. При подаче Запроса на Изменение для решения проблемы также регистрируется номер известной ошибки.

Что представляет собой Запрос на Изменения (RFC)?

Не каждый Запрос на Модификацию обрабатывается как изменение: некоторые повседневные задания, точно определенные и подчиняющиеся

установленным процедурам (стандартизованные), но включающие в себя модификации, могут обрабатываться как Запросы на Обслуживание (например, изменения «категории 0»). В результате возникает следующая классификация изменений:

- Запросы на Обслуживание (здесь: стандартные изменения) - полностью определенные и утвержденные изменения, регистрируемые, но не оценивающиеся Процессом Управления Изменениями. Эти изменения проводятся в рабочем порядке. (Примечание. Не все Запросы на Обслуживание являются изменениями).
- Запросы на Изменения - все другие Запросы на Модификацию инфраструктуры.

Откуда исходят Запросы на Изменения (RFC)?

Запросы на Изменения могут касаться всех аспектов инфраструктуры в пределах сфер действия процессов ITIL. Любой сотрудник, работающий с инфраструктурой, может подать Запрос на Изменения (RFC). Можно определить несколько источников Запросов на Изменения (RFC), например:

- Управление Проблемами - предлагает решения для исключения долговременных ошибок с целью стабилизации предоставления услуг.
- Заказчики - могут запросить больший, меньший Уровень Сервиса или другие услуги. Эти запросы могут подаваться прямо как Запрос на Изменения или направляться через Управление Уровнем Сервиса (SLM) или через Управление Отношениями с заказчиками ИТ (IT CRM).
- Политика компании - тактические и стратегические процессы из области Предоставления услуг (Service Delivery Set) и Указания руководства (Managers Set) могут привести к направлению Запросов на Изменение Услуг. Например, Управление Уровнем Услуг, Управление Доступностью, Управление Мощностями составляют ежегодные планы улучшения услуг, которые позднее могут быть поданы как Запросы на Изменения (RFC).

- Законодательство - если возникают ограничения, регламентирующие бизнес-деятельность, или вводятся новые требования по ИТ-безопасности, непрерывности бизнес-процессов и Управлению Лицензиями.

- Поставщики - поставщики выпускают новые версии и модификации (Upgrades) своих продуктов и сообщают об исправленных ими ошибках. Они могут сообщить, что больше не поддерживают определенные версии или, что не могут гарантировать производительность версии (например, из-за «Ошибки тысячелетия» - Millennium bug). Это может дать толчок Процессам Управления Проблемами или Управления Доступностью к подаче Запроса на Изменения (RFC).

- Проекты - проект часто вызывает ряд изменений. Руководство проекта должно эффективно согласовывать свои действия с Управлением Изменениями с помощью соответствующих процессов, таких как Управление Уровнем Услуг, Управление Мощностями и т. п.

- Любой другой сотрудник ИТ - в принципе, любой сотрудник может подать предложения улучшению услуг. В особенности, ИТ-персонал может способствовать усовершенствованию процедур по поддержке и предоставлению услуг и обновлению руководств.

Регистрация Запросов на Изменения.

Вот примеры информации, которая может включаться в Запросы на Изменения (RFC):

- идентификационный номер Запроса;
- номер проблемы/известной ошибки (если имеется), связанной с Запросом;
- описание и определение соответствующих Конфигурационных Единиц (CI);
- причина изменения, включая обоснование и ожидаемый бизнес-результат;

- текущая и новая версия изменяемой Конфигурационной Единицы;
- имя, адрес и номер телефона лица, направляющего Запрос;
- дата подачи;
- предварительная оценка необходимых ресурсов и времени.

5.5.2 Прием в обработку.

После регистрации Запроса на Изменения (RFC) Управление Изменениями делает первичную проверку, нет ли среди них неясных, нелогичных, непрактичных или ненужных Запросов. Такие Запросы отклоняются с объяснением причин. Сотруднику, направившему Запрос, всегда должна быть предоставлена возможность для защиты своего Запроса.

Проведение изменения ведет за собой обновление в базе данных CMDB, например:

- изменение статуса существующей Конфигурационной Единицы;
- изменение взаимосвязи между различными Конфигурационными Единицами;
- новая Конфигурационная Единица или изменение существующей Конфигурационной Единицы;
- новый владелец или другое месторасположение Конфигурационной Единицы.

Если Запрос на Изменения (RFC) принимается в работу, в регистрационную запись изменения включается информация, необходимая для дальнейшей обработки изменения. Позднее к записи добавляется следующая информация:

- назначенный приоритет;
- оценка степени воздействия и требующихся затрат;
- категория;
- рекомендации Руководителя Процесса Управления Изменениями;

- дата и время авторизации изменения;
- запланированная дата проведения;
- план возврата к исходному состоянию;
- требования по поддержке;
- план проведения изменения;
- информация о разработчике и сотрудниках, ответственных за

проведение изменения;

- фактическая дата и время проведения изменения;
- дата проведения оценки результатов;
- результаты испытания и обнаруженные проблемы;
- причины отклонения Запроса (если необходимо);
- оценка результатов.

5.5.3 Классификация.

После приема Запроса на Изменения (RFC) определяются его приоритет и категория.

- Приоритет показывает, насколько важным является данный Запрос по сравнению с другими. Это, в свою очередь, определяется его срочностью и степенью воздействия. Если изменение касается исправления известной ошибки, код приоритета уже может быть назначен Управлением Проблемами. Однако Управление Изменениями назначает окончательный код приоритета после рассмотрения других обрабатываемых Запросов.

- Управление Изменениями присваивает категорию, исходя из степени воздействия и требуемых ресурсов. Эта классификация определяет дальнейшие процедуры обработки Запроса и обозначает важность изменения.

Определение приоритета.

Пример системы кодирования приоритетов:

- Низкий приоритет - изменение желательно, но его внедрение может быть отложено до более удобного времени (например, до следующего релиза или планового обслуживания).

- Обычный приоритет - нет особой срочности и высокой степени воздействия, но изменение не следует откладывать. На совещании Консультативного комитета (CAB) при выделении ресурсов изменению присваивается обычный приоритет.

- Высокий приоритет - изменение касается серьезной ошибки, затрагивающей ряд пользователей, или новой нетипичной ошибки, затрагивающей большую группу пользователей, или связано с другими срочными вопросами. Такому изменению на ближайшем совещании CAB присваивается высокий приоритет.

- Наивысший приоритет - Запрос на Изменения (RFC) касается проблемы, серьезно влияющей на важнейший для заказчиков сервис, или касается срочного изменения в ИТ (например, новой функциональности для целей бизнеса), срочного изменения законодательства или быстрых небольших изменений, не терпящих отсрочки (Quick fix). Изменения с таким приоритетом классифицируются как «срочные». Для срочных изменений обычные процедуры не используются, так как необходимые ресурсы предоставляются незамедлительно. Может потребоваться проведение срочного совещания Консультативного комитета (CAB) или Руководящего комитета ИТ (IT Steering Committee). Специально для этих целей в компании должен быть сформирован Комитет по срочным изменениям (CAB/EC Emergency Committee) с полномочиями для принятия экстренных решений. Все принятые ранее планы могут быть отложены или прерваны.

Эти коды могут быть представлены в цифрах, например: низкий приоритет = 1 / наивысший приоритет = 4.

Определение категории.

Категории определяются в рамках Процесса Управления Изменениями, в случае необходимости с участием Консультативного комитета (СAB), который определяем степень воздействия изменения и требования, предъявляемые им к ИТ-организации (в первую очередь, выделение ресурсов). Примеры категорий:

- Низкая степень воздействия - изменение, требующее выполнения небольшого объема работ. Руководитель Процесса Управления Изменениями может авторизовать эти изменения без привлечения Консультативного комитета (СAB).

- Существенная степень воздействия - изменение, требующее значительных усилий и оказывающее существенное воздействие на ИТ-услуги. Эти изменения обсуждаются на совещании Консультативного комитета (СAB) для определения необходимых усилий (ресурсов и др.) и потенциального воздействия. Перед совещанием членам Консультативного комитета (СAB) и, возможно, специалистам и разработчикам направляется соответствующая документация.

- Наивысшая степень воздействия - изменение, требующее значительных усилий. Руководителю Процесса необходимо предварительно получить авторизацию на выполнение изменения от руководства ИТ или Руководящего комитета ИТ, после чего изменение представляется на рассмотрение Консультативного комитета (СAB).

Эти коды могут быть представлены в цифрах, например: низкая степень=1/ высшая степень = 3

Большинство изменений относятся к двум первым категориям. В добавление к классификации должны быть также определены группы, участвующие в работе над техническим решением, и услуги, затрагиваемые изменением.

5.5.4 Планирование.

В рамках процесса осуществляется планирование изменений на основе графика, называемого Согласованным планом изменений (Forward Schedule of Change - FSC). План FSC содержит подробную информацию обо всех утвержденных изменениях и их планировании. Члены Консультативного комитета (CAB) дают рекомендации по планированию изменений, так как необходимо учитывать наличие персонала, ресурсов, затраты, различные аспекты задействованных услуг, а также мнение заказчиков. Консультативный комитет (CAB) играет роль консультативного органа. В целом Управление Изменениями имеет делегированные полномочия, т. к. оно действует от лица руководства ИТ. Возможно, что изменения наивысшей категории необходимо утверждать руководством ИТ до их представления Консультативному комитету (CAB). Утверждение изменения имеет несколько аспектов:

- Финансовое одобрение - анализ затрат/выгод и выделение бюджета.
- Техническое одобрение - оценка необходимости, возможности проведения изменения и его степени воздействия.
- Бизнес-одобрение - одобрение пользователями требуемой функциональности приложения и степени воздействия изменения.

Для целей эффективного планирования Управление Изменениями должно взаимодействовать с проектным офисом и другими подразделениями компании, занимающимися разработкой и внедрением изменений. Кроме того, достаточное внимание должно уделяться своевременному осведомлению пользователей о планировании изменений, возможно, в виде рассылки Согласованного плана изменений (FSC).

Политика проведения изменений.

Изменения по разным Запросам можно объединять в одном релизе. В этом случае при неудачной реализации будет достаточно одного возврата к

исходному состоянию (Back-out). Такой групповой релиз должен рассматриваться как одно изменение, даже если он содержит в себе несколько изменений. Релизы могут планироваться с учетом функциональных задач, необходимых для бизнеса. Они могут охватывать аппаратные и программные средства, и их внедрение осуществляется Процессом Управления Релизами. Рекомендуется определить политику компании в этой области и информировать о ней ИТ-организацию и заказчиков. Цель политики - оградить пользователя от ненужного беспокойства («перекапывание дороги каждую неделю»).

После консультаций с участвующими ИТ-подразделениями, Консультативный комитет (СAB) может определить регулярные периоды времени («окна») для проведения изменений, когда степень воздействия на ИТ-сервисы будет минимальной. Подходящими периодами могут быть выходные дни или нерабочее время. Таким же образом могут быть определены периоды, когда допускается минимум изменений или они вообще не допускаются, например, рабочее время или конец финансового года, когда все подразделения пользователей делают отчеты.

Совещания Консультативного комитета (СAB).

Информация о планировании изменений должна распространяться заранее до совещания СAB. Соответствующая документация и информация о пунктах повестки дня также должны рассылаться до совещания.

Повестка дня совещания СAB должна включать ряд постоянных пунктов, в том числе:

- неавторизованные изменения;
- Запросы на Изменения (RFC), которые должны быть оценены членами Консультативного комитета (СAB);
- авторизованные изменения, которые не были представлены на рассмотрение консультативного комитета (СAB);
- открытые и закрытые изменения;

- оценка произведенных изменений.

Оценка степени воздействия и ресурсов.

При оценке необходимых ресурсов и степени воздействия изменения члены Консультативного комитета (СAB), Руководитель Процесса Управления Изменениями и другие участники (определенные Консультативным комитетом) должны учесть следующие аспекты:

- вопросы возможностей («мощности» или «емкости») подвергающихся воздействию услуг;
- надежность и возможность восстановления;
- планы по Управлению Непрерывностью ИТ-услуг;
- планы возврата к исходному состоянию;
- вопросы безопасности;
- степень воздействия изменения на другие ИТ-сервисы;
- регистрация изменения и его предварительное одобрение;
- необходимые ресурсы и затраты (поддержка и обслуживание);
- количество и наличие необходимых специалистов;
- необходимое время на весь цикл изменения;
- новые ресурсы, которые должны быть закуплены и пройти тестирование;
- степень воздействия на текущую операционную деятельность;
- какие-либо возможные конфликты с другими изменениями.

Члены Консультативного комитета (СAB) могут также дать рекомендации по определению приоритета изменения.

5.5.5 Координация.

Об утвержденных изменениях сообщают соответствующим техническим специалистам, которые будут разрабатывать и внедрять эти изменения. Перед внедрением происходит этап тестирования. В разработке, испытании и внедрении утвержденных изменений важную роль может играть

Процесс Управления Релизами. Большое внимание должно уделяться вопросам информирования персонала внутри компании для поддержки изменений.

Подготовка изменения (Building).

Не все изменения проходят отдельную фазу компоновки. Например, стандартные изменения, такие как перемещение персональных компьютеров, могут планироваться и осуществляться незамедлительно.

Компоновка может включать создание новой версии программы с новой документацией, руководствами, инсталляционными процедурами, планом возврата к исходному состоянию и аппаратными изменениями. Управление Изменениями осуществляет контроль и координацию, его поддерживают Процесс Управления Релизами и руководители линейных подразделений, которые обеспечивают предоставление необходимых ресурсов.

Процедура возврата к исходному состоянию должна разрабатываться как часть общей схемы проведения изменения на случай, если изменение не обеспечивает достижение необходимого результата. Управление Изменениями не должно одобрять проведение изменения при отсутствии процедуры возврата. Если изменение влияет на среду пользователя, должен быть составлен коммуникационный план. План внедрения изменения также составляется на стадии компоновки.

Показатели эффективности (Performance indicators) демонстрируют, насколько успешно Процесс Управления Изменениями осуществляет эффективную и рациональную обработку изменений при минимальном возможном отрицательном воздействии на согласованный Уровень Услуг. Эти показатели охватывают такие параметры, как:

- количество завершенных изменений за определенный промежуток времени в разбивке по категориям;
- скорость проведения изменений;
- количество отклоненных изменений;

- количество инцидентов, вызванных изменениями;
- количество возвратов к исходному состоянию, связанных с проведением изменений;
- затраты на произведенные изменения;
- соотношение между расчетными и фактическими затратами ресурсов и времени;
- количество срочных изменений.

Тестирование.

Процедура возврата к исходному состоянию, план внедрения изменения и ожидаемый результат должны проходить тщательную проверку. При этом необходимо учитывать критерии, определенные ранее консультативным комитетом (СAB). В большинстве случаев для испытаний необходима изолированная тестовая среда или лаборатория. Тестирование на ранних стадиях может производиться разработчиками, однако внедрение изменений не может осуществляться без проведения независимого тестирования. Обычно проводится два вида испытаний: приемо-сдаточные испытания для пользователей, при которых представители бизнес-подразделений (обычно заказчики изменения) проверяют его функциональные характеристики, и операционные (эксплуатационные) испытания (Operational acceptance), при которых независимое тестирование проводят те, кто должен поддерживать и обслуживать новую инфраструктуру. Сюда включаются также отделы технической поддержки и Служба Service Desk. Они проверяют соответствующую документацию, процедуры резервного восстановления данных (back-up) и т. д. Необходимы также четкие инструкции для мониторинга качества тестирования и документирования его результатов.

Внедрение (Implementing).

Любой сотрудник соответствующего подразделения, ответственный за администрирование ИТ-инфраструктуры, может получить задание о

непосредственном проведении (внедрении) изменения. Управление Изменениями гарантирует, что это является запланированным изменением. Должен существовать точный план информирования всех вовлеченных сотрудников о проведении изменения (коммуникационный план), например, пользователей, Службы Service Desk, группы администрирования сетей и т. п.

При невозможности проведения необходимого тестирования возможно внедрение изменения для небольшой пилотной группы пользователей и оценка полученных результатов перед внедрением изменения в более широком масштабе.

5.5.6 Оценка.

Необходимо давать оценку произведенным изменениям, за возможным исключением стандартных изменений. При необходимости Консультативный комитет (CAB) принимает решение о проведении последующих дополнительных мероприятий. Должны быть рассмотрены следующие вопросы:

- Привело ли изменение к достижению намеченной цели?
- Удовлетворены ли пользователи результатом?
- Возникали ли какие-либо побочные эффекты?
- Были ли превышены расчеты по затратам и ресурсам?

Если изменение осуществлено успешно, Запрос на Изменение (RFC) может быть закрыт. Это происходит на этапе Анализа результатов внедрения (Post Implementation Review - PIR), т. е. этапе оценки изменения. Если же изменение закончилось неудачно, процесс возобновляется с того места, где он вызвал сбой, с использованием нового подхода. Иногда бывает лучше сделать возврат назад и создать новый или модифицированный Запрос на Изменения (RFC). Продолжение работы с неудачным изменением часто приводит к ухудшению ситуации.

Процедуры с автоматическим отслеживанием времени гарантируют, что этап оценки изменений не будет пропущен. В зависимости от природы изменения оценку можно проводить или через несколько дней, или через несколько месяцев. Например, оценка изменения в используемом ежедневно персональном компьютере может быть совершена через несколько дней, а изменение в системе, использующейся раз в неделю, может быть сделана только через три месяца.

5.5.7 Проведение срочных изменений.

Как бы хорошо ни проводилось планирование, могут быть изменения, требующие наивысшего приоритета. Срочные изменения очень важны для компании и они должны осуществляться как можно скорее. В большинстве случаев на эти изменения направляют ресурсы, предназначенные для других видов деятельности. Срочные изменения могут серьезно повлиять на запланированную работу. Следовательно, задачей является сведение к минимуму числа срочных или неожиданных изменений (с «наивысшим» приоритетом). Возможные превентивные меры включают:

- обеспечение своевременной подачи Запросов на Изменения, пока они не стали срочными;
- при исправлении ошибок, возникших в результате плохой подготовки изменений, возврат не должен заходить дальше прежней версии, то есть дальше Прежнего стабильного состояния (Previous Trusted State). После возврата следует тщательно подготовить новый улучшенный план изменения.

Несмотря на указанные выше меры, срочные изменения все же могут возникнуть. Они требуют процедур для срочной обработки, но с сохранением общего контроля со стороны Процесса Управления Изменениями. В случае возникновения такой ситуации Руководитель Процесса Управления Изменениями может организовать чрезвычайное совещание комитета САВ/ЕС. Если для этого нет времени или если Запрос поступил в нерабочее

время, должен существовать альтернативный способ получения авторизации изменения. Это не обязательно должна быть встреча «лицом к лицу», вместо нее можно провести телефонную конференцию.

Примером является случай, когда экстренный ремонт может быть предложен для разрешения серьезного инцидента. Если дело обстоит очень плохо и отсрочка неприемлема, необходимо следовать процедуре обработки срочного Запроса на Изменение.

Возможна также нехватка времени для проведения нормального тестирования. Например, рабочая станция управляет большой машиной, которая смешивает крахмал для приготовления таблеток в фармацевтическом производстве. Если рабочая станция не будет исправлена в течение часа, крахмал затвердеет, и для его удаления вручную, с помощью молотка и зубила, потребуются работа двух человек в течение двух недель. В это время компания будет терпеть убытки в тысячи долларов за час, так как препараты не будут производиться. При такой ситуации Руководитель Процесса Управления Изменениями должен оценить риски и принять решение о проведении изменения. После этого должны быть пройдены все необходимые этапы нормального процесса для гарантии того, что все пропущенные испытания теперь проведены, вся информация обновлена (произведена регистрация изменений в базе данных CMDB) и что все изменения отслеживаются.

5.6 Контроль процесса.

5.6.1 Отчеты для руководства.

Задачей Процесса Управления Изменениями является достижение баланса между гибкостью и стабильностью. Для характеристики текущей ситуации в организации могут быть использованы следующие отчеты:

- количество проведенных изменений за определенный период времени (всего и по категориям Конфигурационных Единиц);
- перечень причин изменений и перечень Запросов на Изменения;

- количество успешно внедренных изменений;
- количество возвратов к исходному состоянию и их причины;
- количество инцидентов, связанных с проведенными

изменениями;

- графики и анализ тенденций за соответствующие периоды.

Показатели эффективности (Performance indicators) определяют, насколько успешно Процесс Управления Изменениями осуществляет эффективную (Effective) и рациональную (Efficient) обработку изменений при минимальном отрицательном воздействии на согласованный Уровень Услуг. Эти показатели могут быть следующими:

- количество изменений, завершенных за единицу времени, по категориям;

- скорость проведения изменений;
- количество отклоненных изменений;
- количество инцидентов, вызванных изменениями;
- количество возвратов к исходному состоянию, связанных с

изменениями;

- затраты на проведенные изменения;
- количество изменений, осуществленных в рамках расчетных

затрат ресурсов и времени.

5.7 Выводы.

5.7.1 Затраты.

- Затраты на персонал - в большинстве случаев уже имеется персонал, занимающийся координацией изменений. Однако для выполнения задач Руководителя Процесса Управления Изменениями и организации работы Консультативного комитета по изменениям (СAB) возможны дополнительные расходы на персонал. Во многих случаях Управление Изменениями вводится для повышения качества услуг, и возникшие

дополнительные расходы рассматриваются как расходы на качество. После успешного запуска процесса расходы на координацию изменений компенсируются уменьшением расходов на разрешение инцидентов и проблем.

- Затраты на инструментальные средства - расходы на аппаратное и программное обеспечение должны определяться заранее. Часто при внедрении нескольких процессов закупается общее инструментальное средство для Процессов Управления Изменениями, Проблемами, Конфигурациями и Инцидентами. При работе в сложной ИТ-среде невозможно контролировать эти процессы без инструментального средства.

5.7.2 Проблемы.

При внедрении Процесса Управления Изменениями возможно появление следующих проблем:

Работа без средств автоматизации слишком трудоемка, она будет создавать много проблем.

- Возможно сопротивление всеохватывающему Процессу Управления Изменениями, осуществляющему мониторинг всех аспектов ИТ-инфраструктуры. В этом случае необходимо обучение персонала, который должен осознать, что все компоненты ИТ-инфраструктуры могут оказывать значительное влияние друг на друга, и что реализуемые изменения Конфигурации требуют общей координации.

- Возможны попытки проведения изменений в обход согласованных процедур. Абсолютно необходимо, чтобы такие попытки встречали соответствующую реакцию со стороны компании. Целостность Процесса Управления Изменениями зависит от полного соответствия процедурам. Претензии сотрудников и предложения по усовершенствованию Процесса Управления Изменениями должны пониматься и приветствоваться, однако неподчинение необходимо решительно пресекать, иначе весь процесс будет поставлен под угрозу.

Другие способы обеспечения исполнения процедур по Управлению Изменениями включают:

- проведение регулярного аудита, возможно, независимым инспектором, для оценки соответствия процедурам Управления Изменениями;
- осуществление контроля со стороны руководства над внутренним и внешним обслуживающим персоналом и разработчиками;
- обеспечение контроля за всеми Конфигурационными Единицами и версиями программ путем защиты базы данных CMDB и организации регулярного аудита Конфигураций в рамках Процесса Управления Конфигурациями;
- предоставление информации из Управления Инцидентами о фактах доступа пользователей к аппаратному и программному обеспечению, не отраженного в CMDB;
- включение необходимых условий и процедур в контракты с внешними поставщиками;
- назначение на должность Руководителя Процесса Управления Изменениями сотрудника с обширным опытом и достаточными бизнес- (что часто недооценивается) и техническими знаниями. Правильный выбор претендента на эту должность имеет критически важное значение, это не должно упускаться из виду, как часто бывает.

5.7.3 Предложения.

Некоторые проблемы могут быть решены за счет реализации следующих предложений:

- обеспечить, чтобы каждое изменение проходило всю процедуру обработки.
- наладить контакт со всем ИТ-персоналом и всеми поставщиками, чтобы гарантировать их понимание Процесса Управления Изменениями и отказ от попыток проведения изменений без координации;

- обеспечить постоянное проведение окончательной оценки изменений (Анализ результатов внедрения - PIR);
- организовать взаимодействие с Управлением Конфигурациями для гарантированного ввода изменений Конфигурационных Единиц в базу данных CMDB.

6 Глава 4 Оценка эффекта от внедрения СУТ на базе Rational RequisitePro.

6.1 Методика исследования прецедентов окупаемости инвестиций

Компания IDC разработала структурированную методику опроса для сбора качественных и количественных данных по окупаемости инвестиций (return on investment, ROI) клиентов Rational, использующих Rational RequisitePro для управления требованиями.

6.1.1 Определение компонентов ROI

Общий измеряемый экономический эффект (net quantifiable benefit, NQB) представляет собой разницу между экономией, полученной в результате внедрения Rational RequisitePro в сравнении с экономией либо в отсутствии такого решения, либо при внедрении других эквивалентных решений. NQB может быть либо непосредственным, выражаясь через рост доходов или снижение затрат, либо косвенным, выражаясь в экономии ресурсов, которые могли бы потребоваться организации для выполнения задач, возникающих в альтернативном варианте. Все экономические эффекты выражены в долларах США.

Чистые затраты (net cost, NC) представляет собой разницу между общими затратами, связанными с Rational RequisitePro, и аналогичными затратами, связанными с наиболее подходящим альтернативным решением (которое может отсутствовать). Обычно в NC входят расходы на

лицензирование и сопровождение ПО, консультации по продукту, обучение, а также любые дополнительные расходы на оборудование, связанное с внедрением RequisitePro. Все чистые затраты выражены в долларах США.

ROI представляет собой отношение общего экономического эффекта к общим чистым затратам.

6.1.2 Вычисление ROI

ROI представляет собой отношение общего экономического эффекта к общим чистым затратам. Если экономический эффект равен затратам, это отношение равно 1, что соответствует равновесию или отсутствию улучшения. Отношение меньше 1 отражает ухудшение, больше 1 – улучшение.

Показатель ROI часто выражается в процентах, в этом случае значение ROI умножается на 100. Таким образом, процентный показатель ROI вычисляется следующим образом: $ROI = (NQB/NC) \times 100\%$

6.1.3 Другие факторы, влияющие на ROI

ROI является относительным показателем прибыли и затрат, связанных с Rational RequisitePro, в сравнении с альтернативным решением – как правило, тем, которое использовалось организацией для решения данной проблемы бизнеса до внедрения RequisitePro. Отсюда следует, что определения NQB и NC в разных ситуациях будут различны и, в некоторых случаях, результаты могут не иметь четкого количественного выражения.

При выполнении "настоящего" анализа ROI, экономия или доход, выражаемые в человеческих ресурсах, учитываются в анализе только в том случае, если эти ресурсы исключаются в результате внедрения продукта. Однако, так бывает далеко не всегда. Чаще всего ресурсы перераспределяются в те проекты или области, где в них испытывается необходимость. Тем не менее, при этом получается реальная экономия, поскольку в противном случае пришлось бы нанимать дополнительных сотрудников.

В последующих исследованиях прецедентов выяснено, что использование Rational RequisitePro дает такие преимущества, как рост производительности, сокращение цикла разработки, ускорение тестирования и повышенное качество приложений. В случае наличия таких эффектов, проводимый IDC анализ не "требует", чтобы организация реализовала эти преимущества путем сокращения персонала, хотя ожидается, что "лишним человеко-часам" будет найдено продуктивное применение.

6.1.4 Методика сбора данных по ROI

Цель этих опросов заключалась в выяснении хозяйственных проблем, которые пытались решить организации, роли Rational RequisitePro в решении этих проблем и определении ROI, связанного с внедрением RequisitePro.

Для опросов по определению ROI был установлен ряд стандартных вопросов. Цель этих вопросов заключалась как в сборе полных данных по ROI, так и в обеспечении единой структуры для анализа этих данных. Эти вопросы позволили IDC понять общий подход организаций к разработке и развертыванию новых продуктов, а также оценить относительные атрибуты, экономические эффекты и затраты, связанные с Rational RequisitePro.

Использовался трехлетний "горизонт ROI", т.е. все показанные данные отражали накопленную фактическую и планируемую экономию ресурсов за фиксированный трехлетний период с 1999 года по 2001 год.

Весь сбор данных и расчеты ROI в этих исследованиях прецедентов были выполнены аналитиками IDC.

В дальнейшем в этой главе рассматриваются варианты расчёта эффекта от внедрения Rational RequisitePro в процессе управления требованиями и обсуждаются как качественные, так и количественные преимущества, получаемые при внедрении решения на базе Rational.

6.1.5 Исследование прецедента: CitiPower

Чтобы выполнить требования меняющейся нормативной среды и удовлетворить растущие потребности клиентов, компания CitiPower вложила значительные инвестиции в свою ИТ-инфраструктуру.

Обзор и бизнес-среда компании

Компания CitiPower с головным офисом в штате Виктория (Австралия) является полностью подконтрольной дочерней компанией American Electric Power и занимается розничными и оптовыми поставками электроэнергии, в основном – для старой части Мельбурна. Хотя, в сравнении с другими коммунальными службами, контингент заказчиков этой компании может считаться относительно небольшим, эти клиенты достаточно крупны и ценны благодаря своему коммерческому характеру. Приватизация электроэнергетики в Австралии вызвала множество изменений в нормативной среде. Чтобы выполнить требования меняющейся нормативной среды и удовлетворить растущие потребности клиентов, компания CitiPower вложила значительные инвестиции в свою ИТ-инфраструктуру.

Текущая ИТ-среда

В настоящее время штат ИТ отдела CitiPower составляет примерно 100–120 человек. Сюда входит большое количество контрактных сотрудников из таких стратегических партнеров, как IBM, Tier, Halcyon и PricewaterhouseCoopers. Эти контрактные сотрудники были наняты для работы с существующим персоналом в целях реализации компанией CitiPower своей программы розничной конкурентоспособности (Full Retail Contestability, FRC). Эта программа позволяет CitiPower привести себя в соответствие с новыми законами о конкуренции и повысить свою привлекательность для клиентов в новой конкурентной отрасли. Программа FRC является крупномасштабным проектом, который настолько важен для CitiPower, что для его выполнения в соответствии с государственным планом компания решила нанять дополнительный персонал. После реализации

программы будет оставлен небольшой штат сотрудников для поддержки и усовершенствования новой системы.

Новые потребности и роль Rational RequisitePro

Результатом приватизации электроэнергетики стала постоянно изменяющаяся и развивающаяся нормативная среда. Поскольку правительство установило новую структуру правил и норм, коммунальные компании должны быстро адаптироваться к этим изменениям. В связи с этим компания CitiPower инициировала проект по полной реконструкции своей внутренней инфраструктуры. CitiPower использует, среди прочих, такие продукты, как Rational Unified Process, Rational Rose, Rational ClearQuest, Rational Suite Test-Studio, Rational Suite AnalystStudio и Rational RequisitePro, который является интегральным компонентом Rational Suite AnalystStudio.

Продукт Rational RequisitePro позволил централизовать процесс управления требованиями и их отслеживания от стадии обследования до тестирования, что дало неоспоримые преимущества компании CitiPower.

Что касается управления требованиями, продукт Rational RequisitePro позволил централизовать этот процесс и отслеживать их от стадии обследования до тестирования, что дало неоспоримые преимущества компании CitiPower. В начале проекта FRC (сентябрь 1999 г.) CitiPower использовала RequisitePro для сбора требований и отслеживания прецедентов в процессе их анализа. С точки зрения проектной группы, RequisitePro позволяет руководству ИТ отдела понять и контролировать в ходе проекта информацию об исполнителях обновления проекта и авторах версий документации.

Сложность системы FRC убедила руководство компании CitiPower в необходимости значительных инвестиций в создание четких спецификаций системы.

Анализ ROI

Общий измеряемый экономический эффект. Компания CitiPower утверждает, что использование Rational RequisitePro дало экономический

эффект в трех областях. Сложность системы FRC убедила руководство компании CitiPower в необходимости значительных инвестиций в создание четких спецификаций системы. Эти спецификации, по всей вероятности, должны повысить производительность работы на протяжении всего жизненного цикла проекта. Хотя, по оценкам CitiPower, средний рост производительности в процессе сбора требований составит лишь 10%, финансовый эффект будет весьма значительным благодаря масштабу этого процесса. Для сбора требований CitiPower сначала наняла 40 аналитиков. По истечении полутора лет перед завершением сбора требований это число сократилось до 19. Учитывая объем и продолжительность процесса, компания CitiPower считает, что общий экономический эффект от использования RequisitePro в сравнении с ручным методом (использование документов Word, таблиц и электронной почты) составил около 630 тыс. долларов.

CitiPower также уверена, что создание ясных, точных и понятных требований помогло повысить производительность разработчиков и специалистов по обеспечению качества.

Чистые затраты. Затраты по проекту FRC состояли из расходов на лицензирование, поддержку, обучение и консультации. Все эти расходы являлись дополнительными и были связаны с внедрением Rational RequisitePro.

В расходы на лицензирование и поддержку Rational RequisitePro вошла пропорциональная часть соответствующих расходов на Rational Suite AnalystStudio, которая за трехлетний период этого исследования ROI составила почти 9 тыс. долларов. Следует отметить, что RequisitePro является лишь одним из инструментов AnalystStudio, полная стоимость которой составила 26 тыс. долларов. Однако в рамках этого исследования, ограниченного управлением требованиями, распределенные чистые расходы на Rational RequisitePro составили примерно 9 тыс. долларов.

Расходы на обучение 40 аналитиков, использовавших Rational RequisitePro, оказались наиболее значительной расходной статьей проекта FRC и составили около 97 тыс. долларов.

Расходы на консультации, включая поддержку CitiPower со стороны Rational Software, составили 39,5 тыс. долларов.

Общие затраты по проекту FRC в течение трехлетнего периода оказались немногим более 145 тыс. долларов.

Окупаемость инвестиций. Окупаемость инвестиций для этого проекта составила 434%, что соответствует фактору ROI 4,3. Это означает, что на каждый доллар, инвестированный в управление требованиями для этого проекта, была получена прибыль 4,3 доллара.

Преимущества использования Rational RequisitePro

Использование Rational RequisitePro дало компании CitiPower следующие преимущества:

- Централизованный процесс управления требованиями
- Возможность отслеживания требований в фазе тестирования
- Управление шаблонами
- Контроль версий приложений и документации
- Улучшенный контроль проекта
- Повышенная точность требований

Исследование прецедента: проект DaimlerChrysler по созданию профиля технологического пользователя

Обзор и бизнес-среда компании

DaimlerChrysler является международной организацией, занимающейся производством автомобилей, грузовиков и автобусов. Головные офисы объединенного концерна расположены в Штутгарте (Германия) и Оберн Хилз (США, штат Мичиган). Штат концерна насчитывает более 400 000 сотрудников. Группа технического компьютерного центра (ТСС) внутри DaimlerChrysler несет ответственность за оценку информационных технологий и их внедрение в процесс разработки продуктов, помогая

реализовать задачи по обеспечению качества и функциональных возможностей продуктов, их стоимости и сроку выпуска.

Текущая ИТ-среда

DaimlerChrysler обладает чрезвычайно разнородной смесью аппаратных платформ, операционных систем, сред разработки, программных пакетов и пользовательских приложений. Ранее возможности выбора программного обеспечения, удовлетворяющего потребности пользователей, были достаточно гибки, однако сейчас DaimlerChrysler использует для этого более стандартизованный подход в целях обеспечения единообразия ПО, что упростит его сопровождение и использование. Группа разработки приложений Lotus Notes является одной из многих групп ТСС. В нее входит пять разработчиков, по одному на проект. Эта группа одной из первых в ТСС стала использовать RequisitePro.

Новые потребности и роль Rational RequisitePro

ТСС поддерживает операции проектирования транспортных средств концерна Chrysler. Хотя сотрудники группы разработки приложений Lotus Notes успешно справлялись со своими задачами, их методы работы не были максимально эффективны и не обеспечивали тесной межгрупповой координации. Проектные требования собирались эпизодически с помощью встреч, телефонных переговоров, электронной почты и других методов. Выполняемым проектам не хватало четкого плана, для них не устанавливались критические пути развития. Такое отсутствие процесса организованного управления требованиями приводило к несогласованности, срыву сроков и, в конечном итоге, перерасходу бюджета проектов.

Чтобы найти оптимальное решение этих проблем, группа разработки приложений Lotus Notes решила использовать Rational RequisitePro в новом проекте под названием "База данных профилей технологических пользователей" (Technology User Profile, TUP). Проект TUP имеет две основные задачи: 1) идентифицировать внутренних пользователей новых инициатив в области ПО; 2) в сотрудничестве с другими группами ТСС

обеспечить идентифицированных пользователей технологией и инструментами, необходимыми для выполнения их работы.

Используя RequisitePro, группа разработки приложений Lotus Notes намерена устранить препятствия, связанные с координацией действий четырех независимых групп внутри ТСС, которые несут ответственность за поддержку этих внутренних клиентов. С помощью RequisitePro был разработан четкий план и критический путь развития проекта, в рамках которого все четыре группы могут отчитываться о выполнении своих задач. Это открывает возможности для улучшения обмена информацией и управления графиком работ.

В настоящее время базу данных TUP используют около 800 человек и ожидается, что в течение следующих двух лет это число вырастет до нескольких тысяч. Хотя проект TUP еще не завершен, использование Rational RequisitePro уже позволило участникам проекта разработать критический путь развития и создать основной список управляемых требований. Упростился анализ влияния предлагаемых изменений, и появилась возможность распространения информации об изменениях между всеми заинтересованными сторонами. Использование RequisitePro позволило членам проектной группы установить контроль над проектом, который должен оказать положительное влияние на качество продукции, расходы и удовлетворение потребностей клиентов.

Анализ ROI

Общий измеряемый экономический эффект. Концерн DaimlerChrysler считает, что использование RequisitePro дает экономический эффект на трех стадиях жизненного цикла разработки приложений. Размах проекта TUP убедил сотрудников ТСС в том, что уточнение требований должно привести к росту качества приложений, что положительно скажется на производительности пользователей базы данных TUP. Поскольку к концу 2001 года эту базу использовали более 800 человек, то даже небольшой

прирост производительности среди этого контингента пользователей даст ощутимый экономический эффект.

Концерн DaimlerChrysler уверен, что использование RequisitePro существенно повысило производительность работ по сбору требований. Даже по грубой оценке, общий прирост производительности составил около 50%. Однако, поскольку сбором требований занимались лишь два человека в течение четырех месяцев, соответствующий экономический эффект составил лишь 90 тыс. долларов.

Возможности RequisitePro по упорядочению процесса сбора спецификаций также дали эффект в виде ускорения работы 10 экспертов по предметным областям, которые составляли спецификации для аналитиков, занимающихся их сбором. Общий экономический эффект от 50-процентной экономии рабочего времени экспертов составил 28 тыс. долларов.

Возможности RequisitePro по созданию четкого, постоянно обновляемого и понятного репозитория требований также дали дополнительное преимущество, позволив разработчикам улучшить качество своих программ и решений, более полно удовлетворяющих потребности пользователей. Устранение неясности в требованиях, касающихся прецедентов, позволило разработчикам уделить больше внимания самой разработке и повысить качество кода. Руководство ТСС ожидает, что производительность разработчиков возрастет примерно в три раза, что даст общий экономический эффект почти в 126 тыс. долларов.

Более понятные требования также окажут положительное влияние на работу других сотрудников ТСС, связанных с разработкой приложений, например, тех, кто косвенно участвует в поддержке процесса разработки. Ожидается, что рост производительности этих сотрудников составит около 5%, что даст экономический эффект около 30 тыс. долларов.

Последний, но наиболее весомый компонент экономических преимуществ заключается в росте значения полученного приложения TUR, которое будет постепенно повышать производительность конечных

пользователей. Этот эффект обусловлен более детальным пониманием работы и потребностей пользователей, в результате которого база данных и приложение TUP смогут обеспечить для них своевременное получение необходимого аппаратного и программного обеспечения, а также прохождение обучения. Руководство ТСС ожидает, что рост качества приложения TUP будет экономить около одного часа рабочего времени пользователя каждые пять недель. Благодаря большому количеству пользователей TUP и ключевой роли этого приложения, ожидается, что общий экономический эффект будет весьма значительным и составит около 60 тыс. долларов лишь за последние четыре месяца 2001 года.

Общий экономический эффект, связанный с разработкой и использованием приложения TUP, за период исследования ROI составляет около 334 тыс. долларов.

Чистые затраты. Общие затраты по этому проекту оказались исключительно низкими. Затраты на лицензирование были стандартными, составив чуть больше 5 тыс. долларов. Как следствие, расходы на сопровождение оказались почти незначительными, а обучение нескольких аналитиков стоило чуть более 5 тыс. долларов.

Таким образом, общие расходы едва превысили 11 тыс. долларов.

Окупаемость инвестиций. Показатель ROI для этого проекта оказался наивысшим среди всех прецедентов, рассмотренных в этой статье. Причины этого заключаются в комбинации преимуществ, которые распространяются на разработку продукта и оказывают влияние вплоть до конечного пользователя, большого контингента пользователей и достаточно скромного, с точки зрения разработки и относительно своего влияния на пользователей, масштаба проекта.

Окупаемость инвестиций для этого проекта составила 2 980%, что соответствует показателю ROI 29,8. Хотя окупаемость этого проекта уже достаточно высока, в течение следующих нескольких лет он постепенно охватит около 10 000 пользователей, что объясняет высокую

заинтересованность концерна DaimlerChrysler в согласовании усилий по повышению качества приложения TUP.

Преимущества использования Rational RequisitePro

Использование Rational RequisitePro дало концерну DaimlerChrysler следующие преимущества:

- Возможность точного планирования графика работ
- Улучшенное управление требованиями
- Возможность использования требований при определении планов тестирования
- Поддержка управления влиянием, которое оказывают изменения требований
- Проверка планов тестирования и качества продукции
- Разработка критического пути развития
- Улучшенный обмен информацией по проекту

6.1.6 Исследование прецедента: проект DaimlerChrysler

по организации виртуального сервиса

Обзор и бизнес-среда компании

Третий исследуемый прецедент также касается концерна DaimlerChrysler и, фактически, его группы ТСС, задействованной в проекте TUP. Однако в этом случае рассматривается другое подразделение ТСС, чья деятельность связана с виртуальным сервисом.

В проекте VSP создается важная петля обратной связи, которая, как уверены в концерне DaimlerChrysler, должна повысить качество продукции и сократить цикл разработки транспортных средств.

Текущая ИТ-среда

Группа виртуального сервиса в составе ТСС занимается использованием web-технологии под названием "виртуальные инструменты" (Virtual Tools) для разработки сервисных продуктов и принимает участие в проекте виртуального сервиса (Virtual Service Project, VSP). В качестве

первого шага интеграции виртуальных инструментов в процессы разработки технических операций дилеров (Dealer Technical Operations, DTO) был создан продукт под названием Virtual Service Workbench (VSW). Этот продукт предназначен для отслеживания вопросов обслуживания транспортных средств, которые возникают в DTO и направляются в отделы проектирования продуктов и процессов. Таким образом, в проекте VSP создается важная петля обратной связи, которая, как уверены в концерне DaimlerChrysler, должна повысить качество продукции и сократить цикл создания транспортных средств.

Новые потребности и роль Rational RequisitePro

Группа виртуального сервиса испытывала те же трудности при управлении требованиями, что и группа разработки приложений Lotus Notes.

Использование RequisitePro для управления проектом VSW, как надеется группа виртуального сервиса, позволит более строго отслеживать требования по мере их возникновения, обеспечит учет всех требований в фазе разработки и создаст более организованную среду разработки продукта. Проект VSW еще не завершен. Однако группа виртуального сервиса подтверждает, что использование RequisitePro уже улучшило возможности отслеживания и управления требованиями. Планирование стало более точным и прогнозируемым, благодаря четким требованиям упростилась разработка прецедентов, а использование стандартизованного процесса и инструментария расширило возможности группы по переадресации работы в случае возникновения такой необходимости.

По утверждению группы VSP, основным преимуществом от использования RequisitePro стала возможность создания инфраструктуры сбора данных.

Анализ ROI

Общий измеряемый экономический эффект. По утверждению группы VSP, основным преимуществом от использования RequisitePro стала возможность создания инфраструктуры сбора данных. Эта инфраструктура

представляет собой комбинацию методик и инструментов, предоставляемых RequisitePro. Группа VSP обнаружила, что, благодаря RequisitePro, рост эффективности сбора требований составил от 3 до 4 раз. Используя среднее значение 3,5, общий экономический эффект составил около 300 тыс. долларов в расчете на продолжительность процесса сбора требований и число задействованных сотрудников.

Группа VSP также отметила несколько преимуществ RequisitePro, получаемых разработчиками. Интеграция RequisitePro с другими продуктами Rational, включая Rose, упростила создание прецедентов. Однако еще более важно то, что наличие требований к продукту и доступ к ним повысили производительность разработчиков на 40%, поскольку им более не приходится заниматься работой бизнес-аналитиков. Хотя группа VSP работала с RequisitePro менее одного года, общий экономический эффект в расчете на трех разработчиков составил около 54 тыс. долларов.

Итоговая чистая прибыль составила 354 тыс. долларов.

Чистые затраты. В чистые затраты группы VSP вошли несколько узловых лицензий на использование RequisitePro бизнес-аналитиками и значительное количество плавающих лицензий, используемых аналитиками, разработчиками и руководителями. Расходы на лицензирование не амортизировались и, с учетом двухлетней поддержки, составили немногим более 28 тыс. долларов.

Внедрение RequisitePro в группе VSP не потребовало дополнительных расходов на обучение, консультации и оборудование, поскольку сотрудники были уже знакомы с продуктами Rational, и компания недавно провела обновление парка настольных ПК и серверов.

Окупаемость инвестиций. Окупаемость инвестиций для группы VSP составила 1 252%, что соответствует показателю ROI 12,5. Это означает, что на каждый доллар, инвестированный на внедрение RequisitePro в группе VSP, была получена чистая прибыль 12,52 доллара.

Преимущества использования Rational RequisitePro

Группа VSP также выяснила, что преимущества RequisitePro распространяются далеко за пределы сфер бизнес-анализа и разработки. Хорошее знание требований помогло сотрудникам, занимающимся обеспечением качества, в создании сценариев тестирования, однако эти преимущества труднее поддаются количественной оценке. Использование Rational RequisitePro дало концерну DaimlerChrysler следующие преимущества:

Возможность более строгого отслеживания требований

- Создание более структурированной среды разработки продукта
- Рост производительности бизнес-аналитиков и разработчиков
- Упрощение процесса создания сценариев тестирования благодаря наличию более полных требований

6.1.7 Исследование прецедента: Merrill Lynch

Обзор и бизнес-среда компании

Компания Merrill Lynch является одним из мировых лидеров в области финансового менеджмента и консультаций по финансовым вопросам. Управляя активами клиентов общим объемом около 1,6 триллиона долларов, Merrill Lynch предоставляет частным и коммерческим клиентам как банковские услуги по инвестированию средств, так и услуги по управлению инвестициями. Группа технологии работы с частными клиентами (Private Client Technology, PCT) компании Merrill Lynch несет ответственность за управление заказами и их обработку как для местных, так и иностранных индивидуальных клиентов. Технология этих операций очень важна для Merrill Lynch, поскольку она связана с большими объемами выполняемых транзакций и позволяет персоналу эффективно обслуживать потребности своих клиентов.

Текущая ИТ-среда

Отдел обработки заказов, входящий в группу PCT, занимается разработкой приложений обработки заказов. В штат этого отдела входят

постоянные сотрудники, выполняющие задачи управления проектами и бизнес-анализа. Из-за большого объема денежных транзакций, выполняемого Merrill Lynch для своих клиентов, управление заказами является критически важной деятельностью. Задачи, стоящие перед этим отделом, заключаются в создании высокопроизводительных систем, чья доступность, мощность, возможности поддержки и производительность смогут удовлетворять потребностям бизнеса. Руководствуясь этими целями, отдел обработки заказов в прошлом году инициировал проект по замене приложений обработки заказов новым ПО, которое способно расширить возможности существующих систем и повысить производительность.

Новые потребности и роль Rational RequisitePro

Хотя компания Merrill Lynch успешно справлялась с задачами разработки приложений и оперативной работы, было принято решение о необходимости внедрения методики разработки с полным жизненным циклом в целях ускорения создания приложений и повышения их качества. Причинами выбора продуктов Rational стало то, что сотрудники уже имели опыт работы с этой компанией, Rational Unified Process (RUP) зарекомендовал себя с наилучшей стороны, а также тот факт, что Rational предлагает полностью интегрированный набор инструментов, удовлетворяющий потребности всего жизненного цикла разработки. RUP представляет собой процесс, базирующийся на инструкциях, шаблонах и оптимальных методиках, который ускоряет операции разработки ПО и повышает их качество. Хотя Rational RequisitePro не вошел в исходный крупномасштабный выбор внедряемых продуктов, в прошлом году было принято решение о его использовании в проекте обработки заказов. До внедрения Rational RequisitePro для сбора требований использовались документы Word, экранные макеты, совещания и электронная почта.

Однако руководитель группы обработки заказов был уверен, что для повышения скорости и качества работы необходимо установить формальный процесс управления требованиями, допускающий проверку и отслеживание

требований от фазы их определения до тестирования. Таким образом, было решено использовать Rational RequisitePro.

Rational RequisitePro формализовал процесс сбора требований, что положительно сказалось на всех его участниках. Группа обработки заказов также широко использовала RequisitePro для управления требованиями и документацией. Наличие установленных требований позволило обнаружить проблемы и вопросы, которые можно решить на ранних стадиях процесса разработки, не откладывая их до фаз тестирования и производства. Это позволило уменьшить количество проблем при разработке и сократить необходимость в последующей доработке на поздних стадиях процесса или после выпуска продукта. Более того, после окончательного определения требований к интерфейсу пользователя разработчики смогли начать программирование отдельных элементов приложения, пока остальные требования еще рассматривались, что дало возможность организовать псевдопараллельный процесс и ускорило разработку. Поскольку все требования индексировались, участники проекта могли быстро и просто получать доступ к ним и их статусу. И наконец, благодаря хранящимся в Rational RequisitePro требованиям, сценарии тестирования были определены быстрее и более полно, а отдел обработки заказов смог установить точные даты проекта на гораздо более ранней стадии, чем это было возможно прежде. Группа обработки заказов обнаружила, что благодаря этим преимуществам сроки разработки сократились в 2-3 раза при одновременном росте качества приложений.

Анализ ROI

Общий измеряемый экономический эффект. Наиболее весомым аспектом использования Rational RequisitePro для Merrill Lynch стал рост производительности разработчиков, который, по оценкам, составил 35%. Вследствие этого, общий экономический эффект для разработчиков группы PCT составил, по оценкам IDC, почти 580 тыс. долларов.

Хотя этот эффект трудно оценить количественно, группа РСТ также обнаружила, что возросшая конкретность и точность требований привела к росту качества программного кода и уменьшению числа ошибок, которые впоследствии могли бы быть обнаружены в рабочем приложении. Высшее руководство группы РСТ считает – и компания IDC поддерживает это мнение, – что исправление ошибки в рабочем приложении может обойтись в 20-30 раз дороже тех усилий, которые требуются для избежания этой ошибки при точном определении требований. Хотя этот способ позволяет устранить не все ошибки, но, несомненно, многие из них.

Чистые затраты. В отличие от других организаций, получивших рост производительности благодаря предоставляемым RequisitePro возможностям автоматизации сбора требований, для Merrill Lynch использование RequisitePro повлекло за собой дополнительные расходы, связанные с более точным и детализированным процессом, следование которому стало обязательным для бизнес-аналитиков. Хотя результатом этого стала более подробная спецификация, которая привела к росту производительности разработчиков и более качественному приложению, были также и затраты в виде дополнительного времени, потребовавшегося для создания этой спецификации. Группа РТС считает, что два месяца сбора требований привели к дополнительным расходам, составившим почти 42 тыс. долларов.

Расходы на лицензирование и сопровождение RequisitePro составили 42 тыс. долларов и полностью пришлись только на данный проект.

Расходы на обучение, связанные с простым бизнес-аналитиков, составили почти 44 тыс. долларов, а расходы на обучение и консультации, предоставляемые Rational Software, составили около 42 тыс. долларов.

Необходимая для поддержки Rational RequisitePro модернизация ПК и сервера добавила к общим затратам еще более 11 тыс. долларов, а расходы развертывание и резервирование составили немногим более 13 тыс. долларов.

Общие чистые затраты, связанные с использованием RequisitePro в этом проекте, составили около 194 тыс. долларов.

Окупаемость инвестиций. Окупаемость инвестиций для этого проекта составила 298%, что соответствует показателю ROI 2,9.

Возможность получить 300% окупаемости инвестиций при затратах, составляющих более 194 тыс. долларов и не повторяющихся в течение следующих двух лет, говорит о впечатляющем экономическом эффекте, реализованном Merrill Lynch благодаря использованию RequisitePro.

Преимущества использования Rational RequisitePro

Использование Rational RequisitePro дало компании Merrill Lynch следующие преимущества:

- Возможность отслеживания требований вплоть до планов тестирования
- Поддержка управления влиянием, которое оказывают изменения требований
- Рост производительности
- Ускорение выхода продукта на рынок
- Рост качества продукта
- Возможность точного планирования графика работ

6.1.8 Исследование возможного прецедента: проект АСКО

Чтобы выполнить требования меняющейся нормативной среды и удовлетворить растущие потребности клиентов, компания CitiPower вложила значительные инвестиции в свою ИТ-инфраструктуру.

6.2 Выводы

В результате проведенных исследований прецедентов компания IDC выяснила, что использование Rational RequisitePro для управления требованиями дает несомненную и измеримую экономическую выгоду. Хотя при опросе организаций упоминался широкий спектр преимуществ,

измеримый экономический эффект был наиболее очевиден в виде роста производительности труда бизнес-аналитиков. Кроме того, для всех случаев был характерно то, что использование RequisitePro повышало производительность во многих направлениях жизненного цикла разработки ПО, включая разработку и обеспечение качества приложений. Пример проекта TUP концерна DaimlerChrysler также выявил экономические эффекты, распространяющиеся вплоть до внутренних конечных пользователей приложения, которое было разработано на основании требований, управляемых с помощью RequisitePro.

Как показано ниже (Таблица 6), значение показателя ROI, полученного в опрошенных организациях, изменялось от 2,98 до 29,80. Среднее значение ROI для всех четырех прецедентов составило 5,00.

Таблица 6

Прецедент	Общий экономический эффект (\$)	Чистые затраты (\$)	Прибыль на инвестиции (\$)	Показатель ROI
CitiPower	630 018	145 325	484 693	4,34
Daimler TUP	333 706	11 199	322 508	29,80
Daimler VSP	354 000	28 264	325 736	12,52
Merrill Lynch	579 688	194 417	385 271	2,98
Итого	1 897 412	379 204	1 518 208	5,00

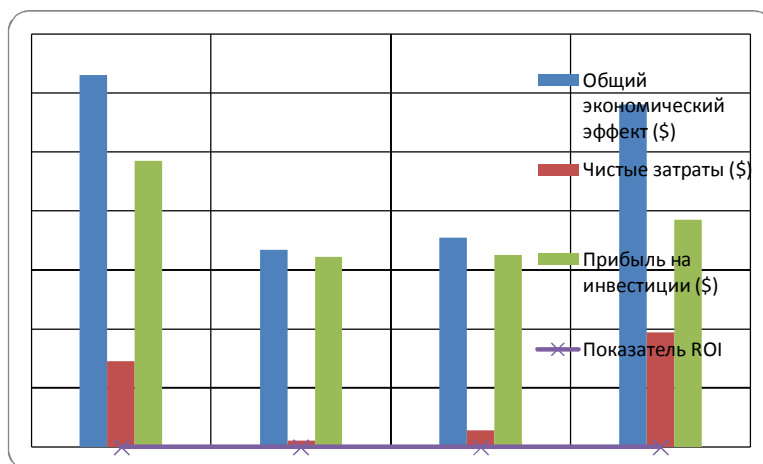


Рисунок 6-1

Основные предположения:

Общий экономический эффект и чистые затраты относятся к основным количественным факторам ROI для всех перечисленных прецедентов.

В данном анализе для расчета ROI использовались временные рамки 1999 – 2001 г.г., хотя некоторые проекты не охватывали все три года.

Расчет текущей стоимости был исключен из анализа ввиду текущего и фактического характера показателя ROI.

Вывод:

Итоговый средний показатель ROI составил 5,00, что отражает высокую привлекательность предложений, связанных с использованием RequisitePro.[29]

Как видно из таблицы Таблица 6, среднее отклонение от среднего значения ROI было достаточно велико. Максимальное значение ROI, полученное в случае проекта DaimlerChrysler TUP, стало в основном результатом относительно большого контингента разработчиков, чья производительность возросла сразу в три раза, а также очень большого контингента конечных пользователей, чей вклад в общий экономический эффект, несмотря на незначительность роста производительности, был достаточно существенен.

Прецедент проекта DaimlerChrysler TUP также указывает на то, что создание петли обратной связи, посредством которой конечный пользователь может влиять на требования, открывает возможности для ускорения жизненного цикла разработки ПО. Это сокращение цикла ускоряет достижение последовательного роста производительности и роста функциональных возможностей приложений, что дает широкие возможности для получения конкурентного преимущества. Хотя расчет ROI, связанного с такой петлей обратной связи, пока не представляется возможным, нетрудно заметить, что преимущества этого метода весьма впечатляют.

С другой стороны, в прецедентах CitiPower и Merrill Lynch значение показателя ROI оказалось в диапазоне 300–400%. В обоих этих случаях уменьшение ROI было в основном обусловлено значительными расходами на обучение, которые, по всей вероятности, существенно сократятся в будущих проектах. Таким образом, CitiPower и Merrill Lynch являются примерами компаний, которые успешно начали внедрение строгого подхода к управлению требованиями и быстро достигли высокой окупаемости инвестиций.

Возможно, наиболее важным и, в какой-то степени, неожиданным результатом, отраженным практически во всех исследованиях прецедентов, явился масштаб экономического эффекта, распространяющегося на весь жизненный цикл разработки ПО в результате внедрения структурированного и интегрированного подхода к управлению требованиями. Значительные инвестиции на ранней стадии жизненного цикла разработки несомненно приносят дивиденды во всех последующих операциях.

Общая окупаемость инвестиций в представленных здесь прецедентах также в высшей степени привлекательна сама по себе. Учитывая, что значение ROI 1 отражает "безубыточность", а значение 2 – 100% прибыли, можно сказать, что представленные в этом документе показатели ROI варьируются от приемлемых до выдающихся.

Несмотря на недостаток внимания, исторически уделяемого управлению требованиями, возрастающая значимость ПО практически в каждом аспекте бизнеса преобразует бывшие когда-то искусством практические методы в науку. Ломая "культурные" барьеры внедрения, существовавшие вокруг управления требованиями, эти прецеденты позволяют организациям сделать ясный и бесспорный вывод, что управление требованиями должно теперь рассматриваться, как краеугольный камень разработки приложений.

7 Заключение.

Поставленная цель данной работы достигнута в результатах решения задачи по автоматизации процесса УТ в проекте АСКО.

Проведен обзор проблематики процесса управления требованиями с описанием известных научно-практических работ в этой области, даны необходимые понятия и определения. Дано развёрнутое описание предметной области с обозначением проблемных мест, приведено функционально-стоимостное сопоставление соответствующих информационных систем, способствующих решению задачи управления требованиями.

Сформирована постановка актуальной прикладной задачи подразделения IBS. Решены следующие методические задачи по управлению требованиями: разработан план управления требованиями для системы АСКО, описывающий процесс УТ, проанализирована структура Проектного Офиса и выделены документы Проекта АСКО, шаблоны которых будут использоваться при управлении требованиями. Представлены EPC и UML – диаграммы, наглядно демонстрирующие бизнес-процесс УТ, как есть (“AS IS”) и как должен стать после автоматизации (“TO BE”). Практические задачи решены путем создания репозитория в RequisitePro для проекта АСКО, Проекта АСКО, введения и размещения документов с требованиями для АСКО, введения глоссария проекта, бизнес-требований к системе, функциональных требований, создания сценариев использования инструмента УТ. Даны рекомендации по тиражированию результатов решения задачи на этапе сопровождения системы, описан процесс управления изменениями требований в терминах ITIL/ITSM. Приведена методика оценки эффекта от внедрения решений на базе продукта IBM Rational RequisitePro, предложенная компанией IDC.

Научная новизна данной работы состоит в совершенствовании существующих методов проектного управления, в частности, управления требованиями, для задач анализа, оптимизации работ с требованиями, обработки информации в результате изменений требований, с целью повышения надёжности и эффективности функционирования автоматизированной системы по формированию консолидированной отчётности, создаваемой подразделением IBS.

Апробация результатов исследования возможна при внедрении решения на этапе сопровождения проекта АСКО.

Основные положения работы опубликованы в сборнике Второй ежегодной научно-практической конференции “Информационные бизнес-системы”.

8 Список использованной литературы

Глава 1:

1. CHAOS, The Standish Group International, Inc., Dennis, MA, 1994, 1997.
2. Computer Industry Daily, December 12, 1997.
3. Dorfman, M. and R. Thayer, Software Engineering, IEEE Computer Society Press, Los Alamitos, CA, 1997 pp. 79.
4. Dorfman, M. and R. Thayer, Software Engineering, IEEE Computer Society Press, Los Alamitos, CA, 1997 pp. 80.
5. Spence, Ian and Leslee Probasco, Traceability Strategies for Managing Requirements with Use Cases, White Paper, Rational Software Corporation, 1998.
6. Rational Unified Process®, Rational Software Corporation, Cupertino, CA, 1999.
7. Дин Леффингуэлл, Дон Уидриг. Принципы работы с требованиями к программному обеспечению. Унифицированный подход. М.: ИД Вильямс, 2002.
8. Jacobson I., Christerson M., Jonsson P., Overgaard G., Object-Oriented Software Engineering – A Use Case Driven Approach, Addison Wesley: ACM Press, 1992.
9. Rational Unified Process® White Paper "Traceability Strategies for Managing Requirements with Use Cases".

Глава 2:

10. Дин Леффингуэлл, Дон Уидриг. Принципы работы с требованиями к программному обеспечению. Унифицированный подход. М.: ИД Вильямс, 2002.
11. Коберн А. Современные методы описания функциональных требований к системам. М.: Лори, 2002.
12. Крачтен Ф. Введение в Rational Unified Process. М.: ИД "Вильямс", 2002.

13. Волков Ю.О. Управление требованиями и автоматизация этого процесса. URL: <http://www.ibs.ru/content/rus/253/2535-article.asp?archive=1> (дата обращения: 30.09.2009).
14. Волков Ю.О. Управление требованиями и автоматизация этого процесса. PC Week/RE (464) 2`2005. URL: <http://www.pcweek.ru/themes/detail.php?ID=69462> (дата обращения: 30.09.2009).
15. Кумсков М.И. ERP для программистов. URL: http://www.it.ru/press_1206.html (дата обращения 14.10.2009).
16. IBM Rational Requisite Pro-2, описание продукта. URL: <http://www.interface.ru/home.asp?artId=10294> (дата обращения: 14.10.2009).
17. IBM Rational RequisitePro evaluators guide. Rational Software. Software group. 2007.
18. Кумар Мани. Новые приемы управления требованиями с помощью Rational RequisitePro. URL: <http://www.interface.ru/home.asp?artId=6465> (дата обращения: 19.10.2009).
19. Галахов И.В., Лапыгин Д.В., Новичков А.Н., Подоляк О.Р., Позин Б.А.. Автоматизированное создание документов серии ГОСТ 34 и 19 с помощью инструментальных средств фирмы IBM Rational. 01.07.2003. Доклад и статья опубликованы в рамках III-ей Всероссийской практической конференции: "Стандарты в проектах современных информационных систем". URL: http://www.caseclub.ru/articles/rup_gost.html (дата обращения: 19.10.2009).
20. Зайдулин Рустам, Новичков Александр. Практическая реализация автоматизированной системы проектирования и формирования нефтегазовой отчетности с использованием IBM Rational RequisitePro и SoDA. URL: <http://anovichkov.msk.ru/?p=37> (дата обращения: 20.10.2009).
21. Бьорн Декер, Эрик Рас, Йорг Рек, Паскаль Жобер, Марко Райт. Wiki в коллективной разработке требований. Открытые системы #03/2007. URL http://www.info-system.ru/software_engineering/software_engineering_wiki.html (дата обращения: 30.09.2009).

22. Домашняя страница А. Коберна URL: <http://alistair.cockburn.us/> (дата обращения: 30.09.2009).
23. <http://twiki.org/cgi-bin/view/Codev/TWikiPresentation2008x08x21> .
24. <http://www.req-labs.ru/archive/conf/15>
25. ГОСТ 34.602-89. Техническое задание на создание автоматизированной системы. Информационная технология. Комплекс стандартов на автоматизированные системы. Дата введения с 01.01.1990.
26. Артефакты дисциплины “Requirements” практики Rational Unified Process.
27. Шаблоны документов практики Rational Unified Process: Requirements Management Plan, Vision, Vision informal, Glossary, Use-Case Model, Context-Free Interview Script, Stakeholder Requests, Supplementary Specification, Iteration Plan, Iteration plan informal, Software Architecture Document, Software Architecture Document informal, Risk List, Supplementary Specifications, Review Record, Software Requirements Specifications, Requirements Attributes.
28. Устав, техническое задание (ТЗ), проектные решения (ПР), соглашения о моделировании, верхнеуровневые схемы проекта АСКО.

Глава 3:

29. Стефан Д. Хендрик, Кэтлин И. Хендрик, Ричард В. Хейман. Окупаемость инвестиций при использовании инструментов Rational для управления требованиями. Информационный документ IDC. 17.03.2003. URL: <http://www.interface.ru/fset.asp?Url=/rational/Rationali.htm> (дата обращения: 19.10.2009).

9 Перечень сокращений

№	Сокращение	Расшифровка
1	АСБУ	Автоматизированная система бюджетного управления ОАО «Газпром»
2	АСКО	Автоматизированная система консолидированной отчетности
3	БНУ	Бухгалтерский и налоговый учет
4	БП	Бизнес- процесс
5	ДБУ	Департамент бухгалтерского учёта ОАО «Газпром»
6	ДО	Дочернее общество
7	ЕВСПД	Единая ведомственная сеть передачи данных
8	ИУС	Информационно-управляющая система
9	ИУС БУиРГ	Информационно-управляющая система бухгалтерского учета и расчетов за газ ОАО «Газпром» на базе системы SAP R/3
10	ИУС П	Информационно-управляющая система предприятия
11	КСЗИ	Комплексная система защиты информации
12	КПЭ	Комплексные показатели эффективности
13	КХД	Корпоративное хранилище данных
14	МСФО	Международные стандарты финансовой отчетности, разработанные Комитетом по Международным стандартам финансовой отчетности
15	НСИ	Нормативно-справочная информация
16	ОБД НСИ	Отраслевой банк данных документарной и фактографической нормативно-справочной информации
17	ОКО	Программный комплекс, используемый для сбора, анализа и контроля данных корпоративной (специализированной) бухгалтерской отчетности Группы Газпром.
18	ОС	Оперативный совет
19	ПМИ	Программа и методика испытаний
20	ПО	Программное обеспечение
21	ПОИБ	Подсистема информационной безопасности
22	ПрО	Проектный офис
23	РП	Рабочий проект
24	РСБУ	Российские стандарты бухгалтерского учета

25	РСПД	Региональные сети передачи данных
26	СБ	Служба безопасности
27	ССКИД	Система сбора и контроля исходных данных
28	СУТ	Система управления требованиями
29	ТЗ	Техническое задание
30	УК	Управляющий комитет
31	УИАС ФЭД	Управления информационно-аналитических систем Финансово-экономического Департамента ОАО «Газпром»
32	ЦОД	Центр обработки данных
33	ЧТЗ	Частное техническое задание

10 Приложение 1 “Постановка задачи на проектирование СУТ”.

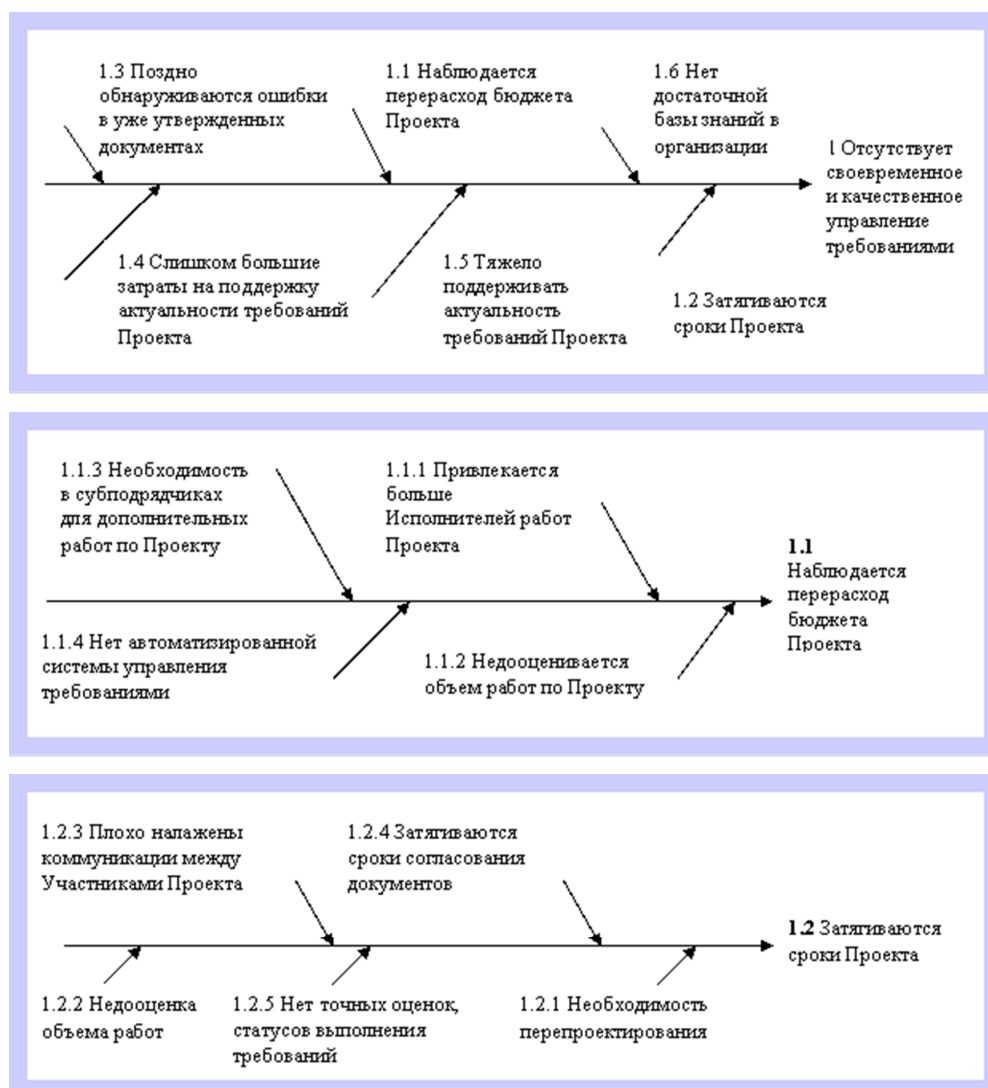
См. отдельный документ [Initial System Request](#).

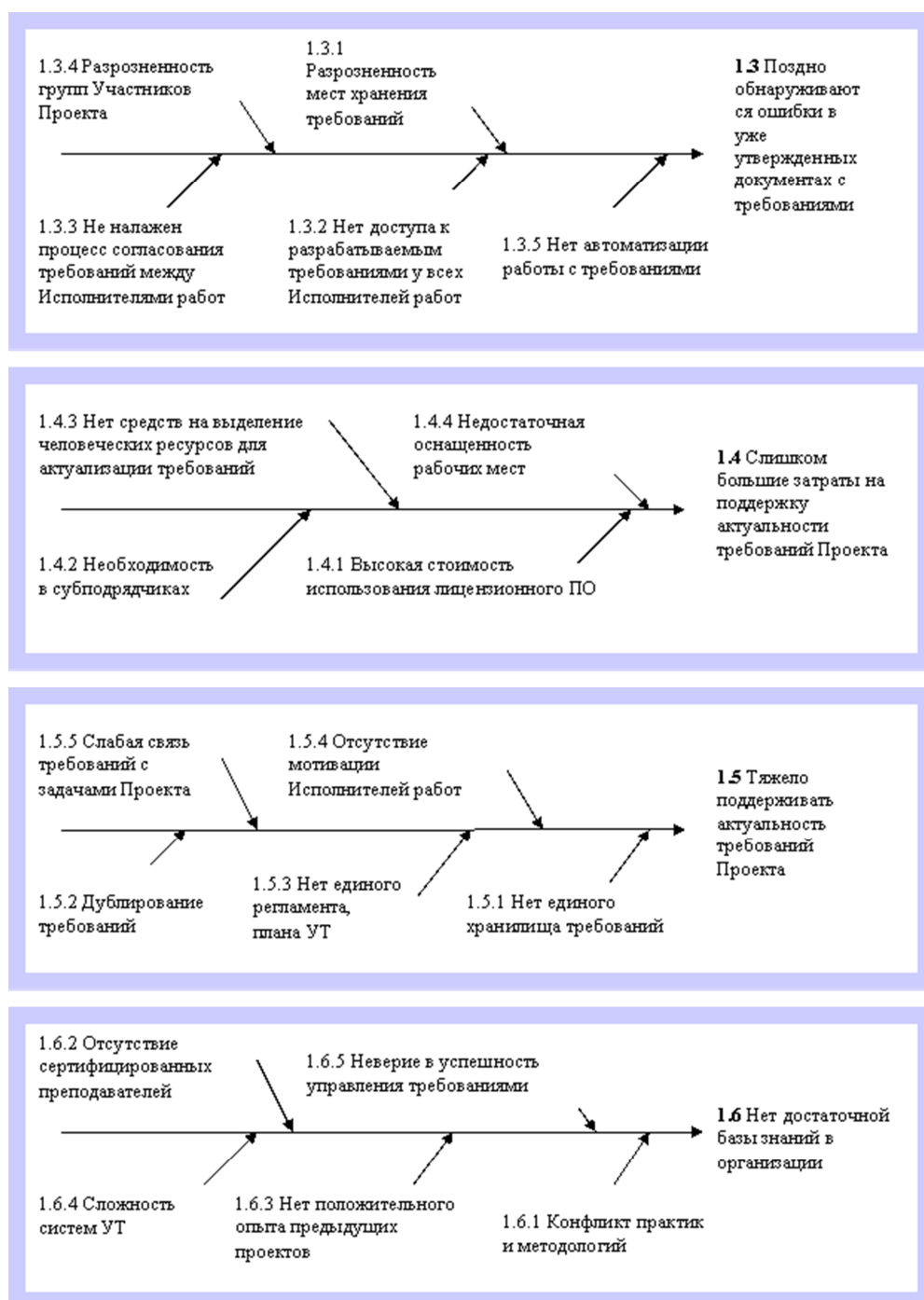
Comment [Irina2]: Проверить работу гиперссылок

11 Приложение 2 “Причинно-следственный анализ”

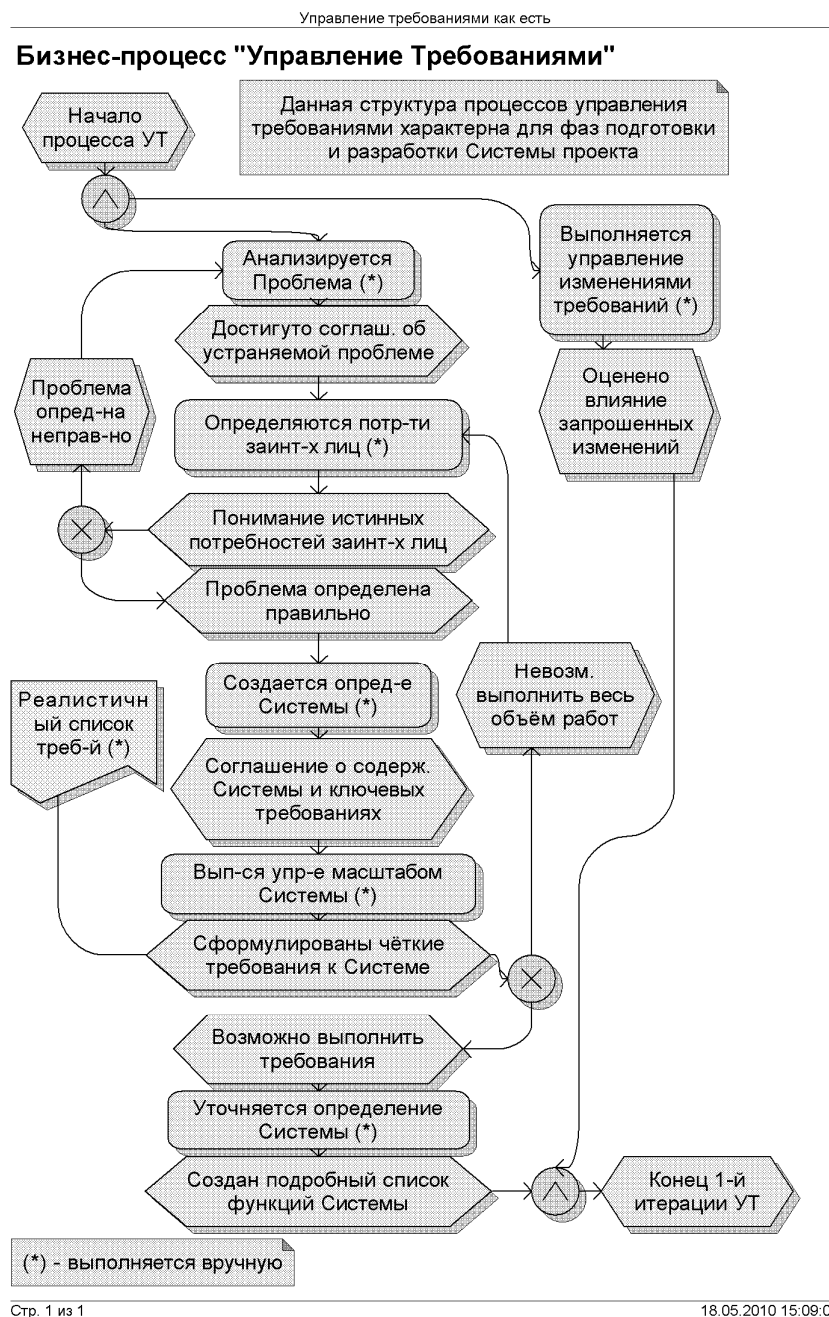
Основная проблема 1.

Причины пронумерованы, учитывая их важность.





12 Приложение 3 “Процесс управления требованиями в проекте АСКО”.

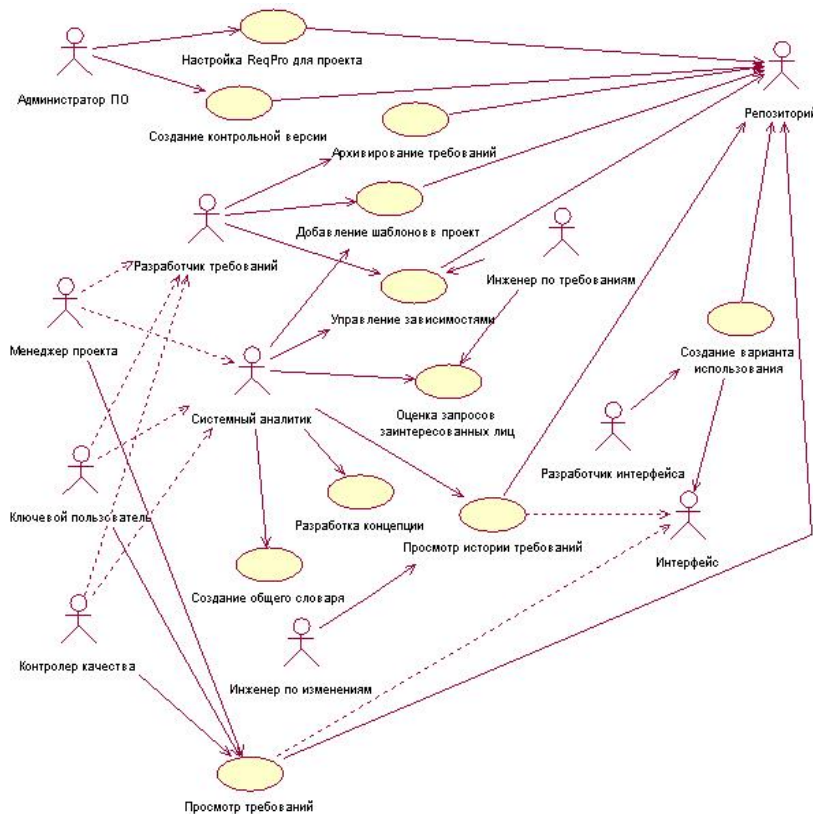


13 Приложение 4 “План управления требованиями”.

См. в отдельном документе [Requirement Management Plan](#).

Comment [Irina3]: Проверить работу гиперссылки

14 Приложение 4 “Сценарии использования СУТ”.



15 Приложение 5 “Репозиторий АСКО. Финальная версия”.

См. отдельный файл