

**Масштабирование тестирования в
организации с гибкими методологиями
продвинутого уровня (СТАL-ATLaS)
Свод знаний к программе
обучения**

v1.0



International Software Testing Qualifications Board

Переведен
Российской Коллегией Экспертов по Качеству ПО
Russian Software Testing Qualifications Board (RSTQB)



Уведомление об авторских правах

Уведомление об авторских правах © International Software Testing Qualifications Board (далее просто ISTQB®) ISTQB является зарегистрированной торговой маркой International Software Testing Qualifications Board.

Авторские права © 2023, авторы перевода 2023 (Илья Кулаков, Александр Александров, Елена Костина).

Авторские права © 2022, Mette Bruhn-Pedersen (Product Owner), Jean-Luc Cossi, Michael Heller, Leanne Howard, Marcelo Chanez, Loyde Mitchell, Iliia Kulakov, and Gil Shekel.

Все права защищены.

Авторы передают свои права International Software Testing Qualifications Board (далее ISTQB). Авторы (владельцы авторских прав в данный момент) и ISTQB (как будущий владелец авторских прав) договорились о следующих условиях использования:

Любое частное лицо или обучающая компания могут использовать эту программу как основу для проведения обучающих курсов, если авторы и ISTQB упомянуты как источник и владельцы авторских прав, при этом в любой рекламе таких курсов данная программа может быть упомянута только после письменного уведомления об аккредитации материалов тренингов коллегий, признанных ISTQB.

Любое частное лицо или группа частных лиц может использовать программу как основу для статей, книг или других производных письменных материалов, если авторы и ISTQB упомянуты как источник и владельцы авторских прав программы.

Любая коллегия, признанная ISTQB, может перевести эту программу (или ее перевод) для других участников.

История изменений

Версия	Дата	Содержание
v1.0	13 мая 2022	Релизная версия
v1.0 RU	27 марта 2023	Перевод на русский язык

Содержание

Уведомление об авторских правах.....	2
История изменений	3
Содержание	4
Благодарности.....	5
0 Введение	6
1 Помощь в обеспечении качества	7
1.1 Что такое помощь в обеспечении качества.....	7
1.1.1 Помощь в обеспечении качества, применяемая у управлению тестированием	8
1.2 Навыки помощи в обеспечении качества	11
2 Совершенствование уровня качества и потока создания ценности в организациях ориентированных на получение ценности - 120 минут.....	15
2.1 Содействие визуализации потоков создания ценности	15
2.1.1 Что такое поток создания ценности.....	15
2.1.2 Визуализация потоков создания ценности	17
2.2 Анализ создания потока создания ценности с точки зрения качества и тестирования ..	22
2.2.1 Метрики для анализа потока создания ценности.....	23
2.2.2 Определение видов деятельности, не способствующих созданию ценности (Потери)28	
3 Непрерывное совершенствование качества и тестирования - 150 минут	33
3.1 Структурированный подход к решению проблем деятельности в областях тестирования и контроля качества.....	33
3.1.1 Цикл Деминга (PDCA)	33
3.1.2 Внедрение цикла Деминга (PDCA) в организации	36
3.2 Системное мышление и анализ первопричин.....	38
3.2.1 Системное мышление	38
3.2.2 Анализ первопричин.....	39
3.2.3 Диаграмма причинно-следственных связей	41
4 Ссылки на материалы	48
5 Дополнительное чтение.....	50

Благодарности

Перевод версии документа 2022 года выполнен Российской Коллегией Экспертов по Качеству ПО Russian Software Testing Qualifications Board (RSTQB): Кулаков Илья, Александров Александр, Костина Елена.

Авторы перевода также выражают благодарность членам Российской Коллегией Экспертов по Качеству ПО Russian Software Testing Qualifications Board (RSTQB), которые приняли участие в комментировании и рецензировании свода знаний к программе обучения Масштабирование тестирования в организации с гибкими методологиями продвинутого уровня (CTAL-ATLaS) (Павел Шариков, Мария Колотева, Андрей Конушин).

Документ был подготовлен командой из International Software Testing Qualifications Board: Mette Bruhn-Pedersen (Product Owner), Jean-Luc Cossi, Richard Green, Michael Heller, Leanne Howard, Ebbe Munk, Francisca Cano Ortiz, Samuel Ouko, Tal Pe'er, Murian Song, Marcelo Chanez, Loyde Mitchell, Iliia Kulakov, Peter Jetter, Salinda Wickramasinghe, and Francisca Cano Ortiz.

Команда благодарит команду редакторов и членов коллегий за их предложения и вклад.

В редактировании, комментировании и голосовании по этой программе участвовали: Ágota Horváth, Ahmed Mohamed Zaki, Andrew Archer, Anna Vitányi, Armin Born, Blair Mo, Chris Van Bael, Chunhui Li, Daniel van der Zwan, Dietrich Leimsner, Florian Fieber, Gary Mogyorodi, Giancarlo Tomasig, Gitte Ottosen, Imre Mészáros, Jing Liang, László Kvintovics, Laura Albert, Li

Chunhui, Marco Hampel, Marton Matyas, Matthias Hamburg, Meile Posthuma, Miroslav Renda, Niels Melin Poulsen, Nishan Portoyan, Ole Chr. Hansen, Paul Weymouth, Péter Sótér, Radoslaw Smilgin, Rik Marselis, Rogier Ammerlaan, Sebastian Małyska, Shujuan Yang, Søren Wassard, Szilárd Széll, Tamás Béla Darvay, Vlad Muresan, and Wim Decoutere.

0 Введение

0.1 Цель этого свода знаний

Этот свод знаний составляет основу программы обучения, используемой для подготовки к международной сертификации продвинутого уровня в области масштабирования тестирования в организации с гибкими методологиями. International Software Testing Qualification Board предоставляет данный учебный план:

1. Членам национальных коллегий для перевода и проведения аккредитации учебных курсов на местном языке. Члены национальных коллегий могут адаптировать программу согласно языковым потребностям и изменять ссылки, адаптируя под местные публикации.
2. Экзаменационным коллегиям, органам по сертификации для составления экзаменационных вопросов на местном языке, адаптируя цели обучения для каждой программы.
3. Поставщикам учебных курсов для подготовки программы обучения и определения соответствующих методов обучения
4. Кандидатам на сертификацию для подготовки к экзамену (как часть учебного курса или независимо)
5. Международным обществам разработки систем и программного обеспечения для повышения квалификации в тестировании ПО и систем, или как основу для книг и статей.

1 Помощь в обеспечении качества – 60 минут

1.1 Что такое помощь в обеспечении качества

Управление качеством объединяет такие дисциплины, как тестирование, обеспечение качества, контроль качества и повышение качества, как указано в программе “Сертифицированный тестировщик программа обучения базового уровня”. Эти дисциплины представляют собой набор мероприятий, способствующих управлению качеством. В данном контексте совершенствование процессов программного обеспечения можно рассматривать как тему, тесно связанную с улучшением качества, которое, в свою очередь, состоит из мероприятий, направленных на повышение качества. Существуют подходы к управлению качеством, которые предполагают использование определенных установок, методов, процессов и инструментов. Эти подходы могут различаться в зависимости от видов деятельности, включенных в управление качеством:

- Традиционное управление качеством программного обеспечения уделяет большое внимание контролю качества и обеспечению качества.
- Всеобщее управление качеством — это один из подходов тестирования в организации с гибкими методологиями. В книге “Иллюстрированный глоссарий по бережливому производству” (Lean Enterprise Institute, 2014) всеобщее управление качеством описывается как подход к управлению, при котором все отделы, сотрудники и менеджеры несут ответственность за постоянное совершенствование качества.
- Помощь в обеспечении качества — это образ мышления и подход к управлению качеством, которое поддерживает гибкость бизнеса. Как и в случае всеобщего управления качеством, деятельности по постоянному совершенствованию уделяется больше внимания, чем деятельности по контролю качества. Переход от контроля качества к помощи в обеспечении качества является фактором успеха для бизнеса (Gartner, 2018). Так же, как в случае всеобщего управления качеством, помощь в обеспечении качества стремится совершенствовать качество, чтобы продукты и услуги соответствовали или превосходили ожидания клиентов. Это означает, что помощь в обеспечении качества способствует созданию организации, ориентированной на получение ценности.

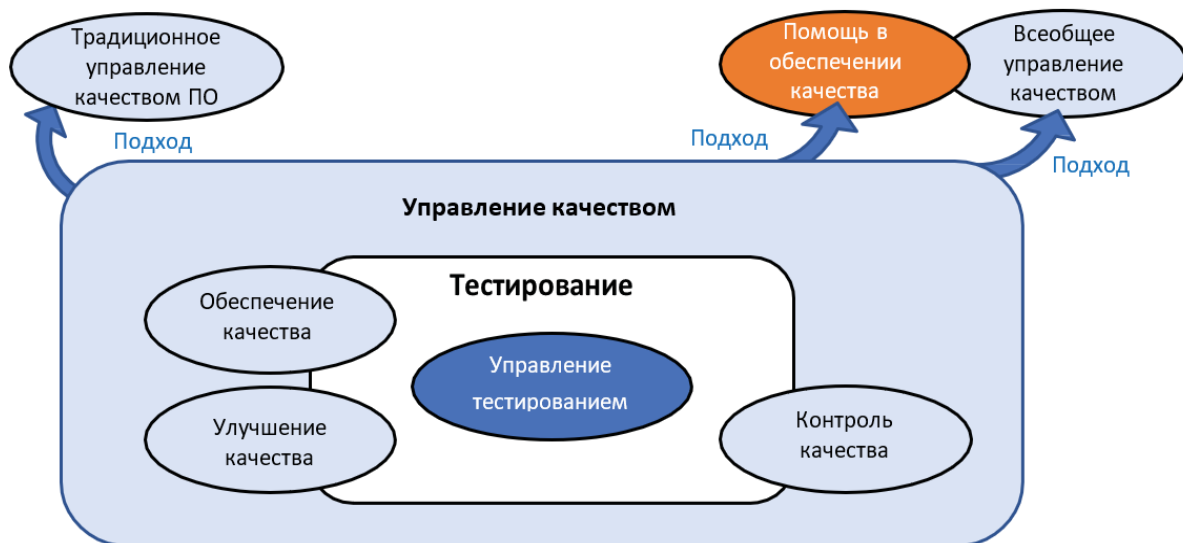


Рисунок 1.1 Помощь в обеспечении качества как подход к управлению качеством

Как видно из рисунка 1.1, многие практики и подходы пересекаются.

1.1.1 Помощь в обеспечении качества, применяемая к управлению тестированием

Гибкие методы управления тестированием опираются на традиционные приемы и техники управления качеством программного обеспечения и сочетают их с новым мышлением, культурой, поведением, методами и приемами, полученными в результате применения подхода **помощи в обеспечении качества**. Взаимосвязи приведены на рис. 1.2. Решение о том, какой аспект следует включить из каждого подхода, сильно зависит от контекста

Однако, если организация стремится повысить гибкость своего бизнеса, то принятие подхода **помощь в обеспечении качества** поддержит это направление.

Традиционное управление тестированием имеет тенденцию фокусироваться на управлении и контроле работы других. Управление тестированием в гибкой организации имеет более широкий охват, чем просто сосредоточение на тестировании программного обеспечения. Переводя гибкое управление тестированием на подход **помощь в обеспечении качества**, лидеры гибкого тестирования тратят больше времени на предоставление и расширение возможностей другим, чтобы те могли самостоятельно управлять тестированием. Цель этой поддержки состоит в том, чтобы способствовать совершенствованию навыков организации в области тестирования и обеспечения качества с целью создания лучшего межфункционального сотрудничества в команде.

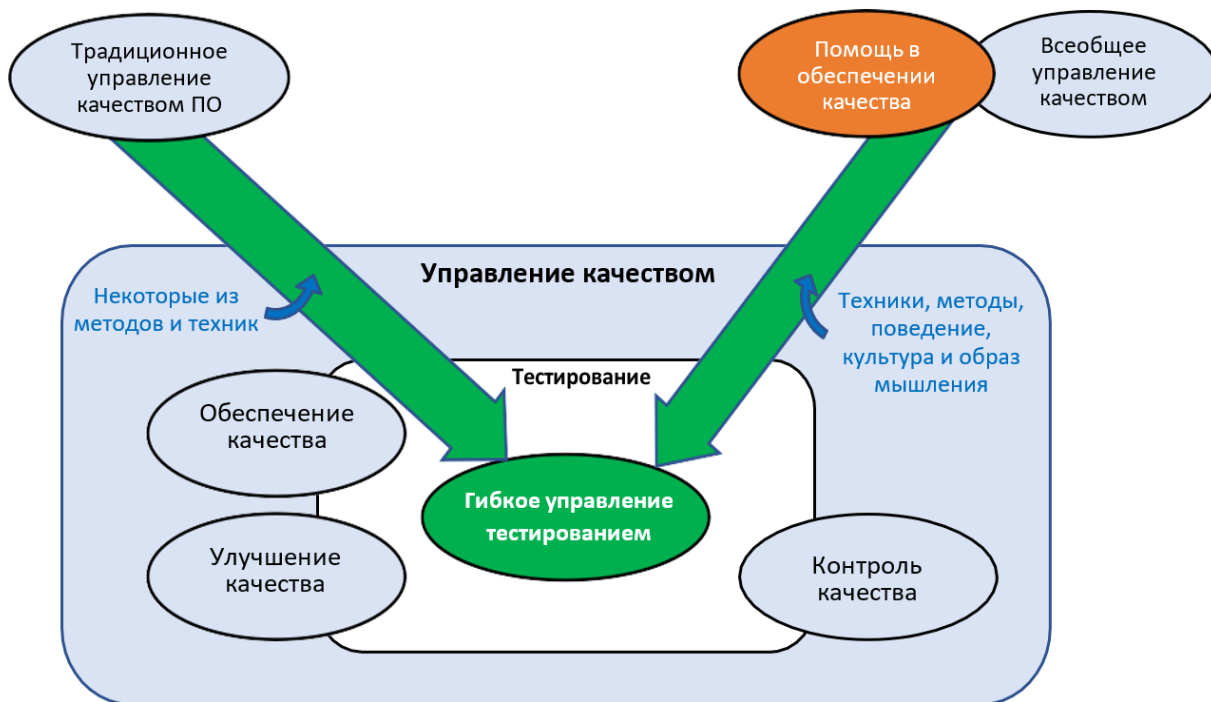


Рисунок 1.2 Гибкое управление тестированием объединяет подходы

Гибкость бизнеса также способствует переходу от традиционных управленческих ролей к самоуправляемым командам реализации проекта и стимулирующим к работе менеджерам. Как следствие, руководители проектов и руководители тестирования иногда с трудом находят свое место в организациях, которые стремятся к гибкости бизнеса. Подобный сдвиг означает, что люди, исполняющие традиционные роли руководителя тестирования, координатора тестирования, инженера по обеспечению качества и тестировщика, должны уделять больше времени и прилагать больше усилий к развитию необходимых навыков и компетенций, связанных с управлением качеством во всей организации, вместо того чтобы фактически производить всё тестирование самим.

Благодаря гибкости бизнеса для оптимизации качества и потока вместо поиска дефектов происходит движение в сторону их предотвращения. Автоматизация, сдвиг влево, непрерывное тестирование и другие мероприятия по обеспечению качества необходимы, чтобы угнаться за инкрементальными поставками организаций, ориентированных на клиентов. Эти практики часто описываются с использованием концепции, называемой "встроенным качеством". Кроме того, появляется возможность применения "=". Практики и деятельность в формате сдвиг вправо сосредоточены на наблюдении и мониторинге решений в операционной среде и измерении эффективности этого программного обеспечения в достижении ожидаемых бизнес-результатов. Подобные практики часто описываются с использованием концепции наблюдаемости.

Переход к подходу **помощь в обеспечении качества** предоставляет множество возможностей для движения в сторону того, чтобы качество считалось коллективной ответственностью во всей организации. Одна из возможностей заключается в том, чтобы руководство организации

поддерживало сотрудничество в рамках групп экспертов, часто известных как профессиональные сообщества. Основной целью экспертных групп должно быть сотрудничество с командами по реализации проекта для распространения знаний и поведенческих норм. Успешное внедрение подхода помощь в обеспечении качества к управлению качеством приводит к:

- Разработке организацией непрерывного подхода к качеству, уделяя особое внимание вопросам совместной работы по обеспечению качества и автоматизированного тестирования
- Меньшему количеству передач задач для тестирования, которые замедляют реализацию ценности
- Меньшей зависимости от тестирования на поздних этапах процесса поставки, уменьшая тем самым общую стоимость затрат на качество

Существует еще много других положительных результатов помощи в обеспечении качества, которые будут рассмотрены в последующих главах.

1.2 Навыки помощи в обеспечении качества

Лидер гибкого тестирования и все другие руководители в организации с гибкими методологиями должны развивать навыки, необходимые для формирования мышления и культуры, ориентированных на качество. Это означает развитие как компетенций команды по доставке ценности, так и общего понимания потоков создания ценности и совершенствования практик тестирования.

Лидеры гибкого тестирования используют такие навыки, как коучинг качества, фасилитация, тренинги и лидерство изменений.

Примеры:

1. Команде, работающей по гибким методологиям, может потребоваться помощь, чтобы понять, как реализация ее проекта интегрируется с реализацией других команд, чтобы обеспечить окончательное решение. Лидер гибкого тестирования может помочь организовать семинар по обучению технике визуализации потока создания ценности с участниками из разных команд, сначала продемонстрировав технику, а затем обучив команду, задавая вопросы о различных этапах потока создания ценности (см. следующую главу).
2. Членам команды требуется помощь в совершенствовании работы в стрессовых ситуациях, поскольку обнаружилось, что в подобные периоды увеличивается количество дефектов. Лидер гибкого тестирования может научить команду сосредотачиваться на качестве.

Некоторые дополнительные задачи, к которым может быть привлечен лидер гибкого тестирования, включают:

- Помощь в создании культуры качества и тестирования
- Предоставление рекомендаций, вдохновения и мотивации инженеров для совершенствования их знаний и навыков в области качества и тестирования
- Популяризацию достоинств и преимуществ разработки через тестирование (TDD) и разработки на основе поведения (BDD) (встроенное качество)
- Визуализацию влияния тестирования и качества на корпоративном уровне
- Общение с лицами, заинтересованными в продуктах и решениях
- Защиту интересов клиентов

У лидеров гибкого тестирования есть много возможностей помочь сотрудникам развить свои компетенции. Это может реализовываться в виде коротких тренингов для решения конкретной проблемы или в виде небольшой серии практических занятий в рамках повседневной работы. Часто ситуация возникает незапланированно, и лидеру гибкого тестирования просто нужно идентифицировать её в тот момент, когда она возникает, и провести работу с отдельным сотрудником или командой. В других ситуациях лидер гибкого тестирования может создавать коучинговые и обучающие группы с практикующими специалистами или экспертами. Эти группы могут помочь членам команды понять, что существует необходимость узнать больше о некоторых компетенциях или об их значимости для доставки ценности. Изменение культуры и мышления в организации может потребовать значительных и непрерывных усилий по коучингу и управлению изменениями в течение длительного периода времени. Поэтому работа лидера гибкого тестирования существенно отличается от работы руководителя тестирования.

Лидер команды тестирования, работающей по гибким методологиям, может оказать качественную помощь в команде, отвечающей за доставку ценности, в то время как лидер гибкого тестирования уделяет больше внимания для совершенствования качества организации в целом.

Лидерство в изменениях

Организации, которые хотят успешно перейти к гибкости бизнеса, должны иметь сотрудников, обладающих навыком лидерства в изменениях, для инициации деятельности по облегчению управления изменениями. Принятие подхода помощи в обеспечении качества обеспечивает поддержку всех членов команды и всей организации в выявлении возможностей и угроз, проведении экспериментов и реагировании на изменения. Помощь в обеспечении качества должна соответствовать программе управления изменениями на корпоративном уровне. Существует множество различных моделей стимулирования изменений, например, 8-шаговый процесс ведения изменений (Kotter, 2012), модель ADKAR индивидуальных изменений (Prosci Inc., н.д.) и цикл Деминга (Lean Enterprise Institute, 2014).

Важно учитывать человеческий фактор, когда эмоции влияют на способность справляться с изменениями. То, как с этими эмоциями справляются, играет важную роль в успешном внедрении изменений. Изменения дают сотрудникам возможность расти, и поэтому управление изменениями должно учитывать различные стили и темпы обучения.

Управление изменениями с течением времени требует постоянной адаптации к организационным факторам и к неустойчивости рынка. Оно также требует баланса между управлением по вертикали “сверху вниз” и управлением по вертикали “снизу вверх”, чтобы обеспечить сотрудникам возможность вносить изменения.

Помощь в обеспечении качества помогает находить улучшения, способствуя тому, что в бережливом производстве называется кайдзен, а в рамках Nexus называется ретроспективной спринта (Scrum.org, 2021). Лидеры гибкого тестирования и лидеры команд тестирования, работающих по гибким методологиям, влияют на изменения, используя свои навыки руководства изменениями, работая с другими заинтересованными сторонами для продвижения к помощи в обеспечении качества и участия в визуализации потока создания ценности. Важная часть управления изменениями состоит в том, чтобы сделать изменения наглядными и отметить достижения.

Вот некоторые примеры:

- Отстаивание тестирования компонентов для правильного охвата тестирования и методологии “сдвиг влево”
- Содействие созданию библиотеки автоматизированных сценариев, чтобы команды могли делиться этими ресурсами между собой, способствуя повторному использованию
- Внедрение в организации инструментов, которые интегрируют, обеспечивают наглядность и синхронизируют информацию.

Коучинг качества

Как и другие формы коучинга, коучинг качества — это форма диалога между коучем и одним или несколькими обучаемыми. Коучинг качества фокусируется на выявлении и решении проблем, связанных с качеством, потоком деловой ценности и сотрудничеством с клиентами.

Коучинг фокусируется на том, чтобы помочь людям осознать свои ценности, страхи и ограничивающие убеждения, которых они могут придерживаться. Поэтому коучинг важен в

организациях, которые претерпевают значительные изменения, такие как переход от классической организации, управляемой программами и проектами, к организации приближающейся к принципу гибкого бизнеса.

Это являлось, и в какой-то степени до сих пор является общим подходом или принципом коучинга, согласно которому обучаемый человек неявно знает решение конкретной проблемы и что роль коуча состоит в том, чтобы помочь обучаемому человеку осознать это и, следовательно, прийти к решению. Но коучинг также может осуществляться как более тесный диалог между человеком, которого тренируют, и коучем. В тесном диалоге меньше внимания уделяется достижению цели или решению проблемы, и больше внимания уделяется пониманию и осмыслению.

Подобный диалог требует, чтобы коуч и лицо (лица), которого (которых) тренируют, были готовы участвовать в разговоре и анализировать сказанное. Коуч может поставить себя на место обучаемого человека, чтобы понять его точку зрения, а затем связать её со своей точкой зрения и позицией во всем, что они исследуют.

Коучинг качества - важный навык при работе с совершенствованием качества. Некоторые события и гибкие процессы очень хорошо подходят для тесного диалога, например, ретроспективы. В зависимости от ситуации может потребоваться дополнить существующие процессы гибкими процессами, посвященными коучингу качества.

Коучинг качества также может использоваться вне командных мероприятий на индивидуальной основе, например, когда вы вместе с сотрудником изучаете новый навык.

Важно создать безопасное пространство для обучаемого, так как коучинг качества может затронуть фундаментальные ценности и ограничивающие убеждения человека.

Фасилитация

Фасилитация — это навык, используемый для того, чтобы помочь в достижении результата или решения, поддерживая отдельных людей посредством взаимодействий. Задача фасилитатора состоит в побуждении коллег целенаправленно использовать их отличительные знания и навыки для этой цели.

Фасилитация является важным навыком в оказании помощи в обеспечении качества, поскольку она позволяет каждому участвовать в дискуссиях о качестве и брать на себя ответственность за решение проблем в вопросах качества. При традиционном подходе к управлению тестированием специалисты по обеспечению качества и тестированию более склонны делиться с другими тем, что им нужно сделать для решения проблем качества. Впоследствии они отслеживают и контролируют наличие совершенствований. В организации с гибкими методологиями все члены команды разделяют ответственность за встроенное качество. Крайне важно, чтобы руководитель гибкого тестирования мог вовлекать различных участников в процессы и беседы об совершенствовании качества и позволять другим находить и внедрять решения проблем качества.

Тренинг

Существует множество различных методов обучения, например, в классе или дистанционно, самостоятельно на рабочем месте, в формате ситуационных семинаров, групповых дискуссий, а также наставничество, стажировка и обучение по принципу “взаимного обучения”. Важно, чтобы руководитель гибкого тестирования мог разработать различные методы обучения, с помощью которых каждый сотрудник мог приобрести необходимые знания и навыки. Важной тенденцией является микрообучение, когда появляется возможность добавить короткие учебные сессии в любое время.

Чтобы действительно масштабировать обучение, руководитель гибкого тестирования может объединиться со службой персонала, сосредоточив внимание на обучении и развитии талантов. Тренинг, который помогает людям совершенствовать свои навыки, может использовать такие методы, как стажировку и обучение на рабочем месте. Сотрудничество со службой персонала в данном контексте может принести большую пользу.

2 Совершенствование уровня качества и потока создания ценности в организациях, ориентированных на получение ценности — 120 минут

Как упоминалось в разделе 0.9 «Бизнес-контекст» Учебного плана «Масштабирование тестирования в организации с гибкими методологиями» продвинутого уровня», при следовании принципу гибкости организации комбинируют правила, структуры, методы, процессы и практики, полученные в рамках различных дисциплин или подходов. Многие предприятия сосредотачиваются на определении ценности, которую они создают, и организации процессов по оптимизации потоков создания ценности. Описанные активности помогают эффективному созданию ценности для клиента в условиях растущей нестабильности и изменчивости мира.

2.1 Содействие визуализации потоков создания ценности

Качество и тестирование — важные аспекты, которые следует учитывать при определении и оптимизации потоков создания ценности как в ходе операционной деятельности, так и в процессе разработки (подробнее см. п. 2.1.1 «Что такое поток создания ценности?»). Поэтому важно, чтобы сотрудники, ответственные за тестирование, а также все те, кто вносит вклад в поток создания ценности, имели четкое понимание концепции и образа мышления, лежащих в основе потоков создания ценности, как предусмотрено методологией бережливого производства.

Бережливое мышление и практики бережливого производства сосредоточены на максимизации результата создания ценности путем пересмотра всей системы или потока создания ценности от начала до конца. Этот подход отличается от анализа элементов потока создания ценности по отдельности, который приводит к локальной оптимизации, например, только в одной функциональной сфере. Локальная оптимизация способна привести к снижению общего результата создания ценности и, следовательно, лишь к частичной оптимизации потока создания ценности. В организациях, ориентированных на получение ценности, занятые в сфере качества и тестирования сотрудники помогают оптимизировать поток создания ценности целиком, а не только в части тестирования.

2.1.1 Что такое поток создания ценности

Поток создания ценности — это группа или совокупность рабочих шагов, в число которых входят задействованные сотрудники и системы, а также информация и материалы, используемые в рамках рабочих шагов. Каждый рабочий шаг должен добавлять ценности предыдущим шагам, а все рабочие шаги вместе — формировать поток создания ценности для клиентов.

Поток создания ценности начинается с идей сотрудников, потребностей клиентов или проблем, требующих решения. Сотрудники, занятые в рамках потока создания ценности, организуют и структурируют рабочие шаги в его составе, чтобы эффективно разработать для клиента продукт или решение. Для минимизации деятельности, не способствующей созданию ценности, поток следует постоянно оптимизировать.

Все потоки создания ценности включают меры по обработке информации от клиента и по преобразованию продукта на его пути от производства к клиенту. Поскольку тестировщики должны глубоко понимать сферы деятельности клиента, им часто доступны инструменты, позволяющие

определить точки соприкосновения для коммуникации с клиентом и понять, как получаемая от клиентов информация влияет на поставку или разработку продукта.

Каждый участник команды, работающей по гибким методологиям, должен иметь доступ к клиенту, чтобы суметь внести вклад в совершенствование потока создания ценности. Принцип бережливого мышления предполагает обращение с каждым сотрудником организации, для которой вы разрабатываете продукт, как с клиентом. Если прямой контакт с ними невозможен, можно попробовать найти других представителей клиента.

Потоки создания ценности могут относиться к сфере операционной деятельности и могут быть связаны с разработкой.

Потоки создания ценности в сфере операционной деятельности — это совокупность рабочих шагов и сотрудников, задействованных в производственном процессе, от размещения заказа до релиза (Lean Enterprise Institute, 2014). Например, в службе обмена сообщениями оператора связи такой процесс содержит пять рабочих шагов, начиная с клиентской подписки и заканчивая доставкой сообщения. Он представлен в виде схемы на **рисунке 2.1**.

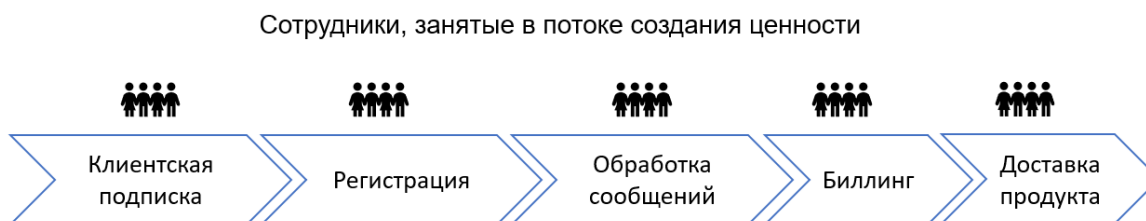


Рисунок 2.1 Пример службы обмена сообщениями

Потоки создания ценности, связанные с разработкой, предполагают работу с продуктом от концепции до вывода на рынок (Lean Enterprise Institute, 2014). Данный процесс представлен в виде схемы на **рисунке 2.2**.

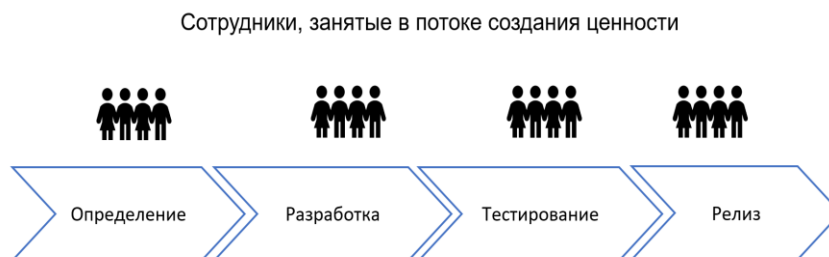


Рисунок 2.2 Пример потока создания ценности, связанного с разработкой (упрощенный)

Иногда потоки создания ценности в сфере операционной деятельности и связанные с разработкой могут совпадать, например, для компании, которая занимается разработкой и поставкой ИТ-решений. Лидеры гибкого тестирования участвуют в определении и анализе потоков создания ценности.

В рамках процесса помощи в обеспечении качества необходимо также помогать остальным участникам взглянуть на тестирование и качество продукта в более широкой перспективе. Участвуя в определении и анализе потоков создания ценности, лидеры гибкого тестирования совершенствуют и качество, и сам поток ценности.

Если работа по определению и описанию потоков создания ценности уже выполнена, следующим шагом является их анализ для оптимизации качества и самого потока (подробнее см. п. 2.2 «Анализ потока создания ценности с точки зрения качества и тестирования»). Если описание потоков создания ценности отсутствует или является высокоуровневым и требует дополнительных данных, тестировщики и специалисты по обеспечению качества могут содействовать процессу визуализацией потока создания ценности (см. п. 2.1.2 «Визуализация потока создания ценности»).

2.1.2 Визуализация потоков создания ценности

Визуализация потока создания ценности (Value Stream Mapping, VSM) — это техника визуализации и анализа рабочих шагов в рамках потока создания ценности, включая поток рабочих элементов (материалов) и информацию, необходимую для производства продукта или услуги. Такой подход позволяет проанализировать:




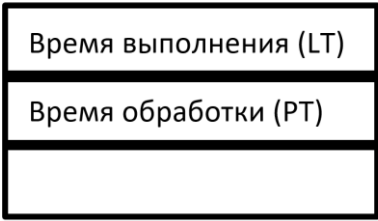

- деятельность, способствующую созданию ценности
- деятельность, не добавляющую ценности, однако необходимую
- деятельность, не добавляющую ценности и не являющуюся необходимой (потери)

Помогает ли деятельность созданию ценности, определяется с точки зрения клиента. Есть виды деятельности, не добавляющие ценности с точки зрения клиента. Некоторые из них необходимы компании для разработки и релиза продукта, например, системное тестирование. Другие можно исключить или сократить без ущерба конечному продукту.

При первом использовании техника визуализации потоков создания ценности обеспечивает высокоуровневую визуализацию текущего статуса и аналогичную визуализацию желаемого статуса в будущем. Кроме того, в процессе определяются инициативы по развитию, необходимые для перехода от текущего статуса к желаемому.

Преимущество техники визуализации потоков создания ценности заключается в возможности оптимизировать поток создания ценности, что достигается путем непрерывного совершенствования мер по созданию ценности и, в частности, исключения деятельности, не добавляющей ценности или изменения её структуры. Поскольку низкое качество влечет корректировки продукта и задержки релиза, техника визуализации потоков создания ценности позволяет совершенствовать качество потока создания ценности в целом. Также этот способ дает общее представление о необходимых объеме и скорости потока создания ценности для удовлетворения клиентского спроса. Потоки создания ценности в сфере разработки тесно связаны с конвейером непрерывной поставки. Однако при разработке программного обеспечения не так легко дать количественную оценку, как при материальном производстве, поскольку программное обеспечение постоянно модифицируется. Это относится к потребностям или требованиям (входные параметры), работе, которую необходимо проделать, чтобы перейти от набора задач продукта к инкременту продукта (правила преобразования), самому инкременту продукта (выходные параметры) и рынку, на котором выходит релиз инкремента продукта (результат). Наконец, визуализация потока создания ценности способна повысить видимость и понимание того, как совместная работа сотрудников, команд и специалистов поддерживает и, следовательно, улучшает сотрудничество.

Техника визуализации потоков создания ценности использует различные условные обозначения. Метод первоначально использовался для анализа и совершенствования производственных систем, но позднее был адаптирован для работы в других сферах, например, разработке ПО и продукции. В качестве отправной точки предлагается использовать простые обозначения, подходящие для разработки услуг или продуктов. См. пример на **рисунке 2.3**.

	<p>Рабочий шаг или процесс.</p>
	<p>Переход разрабатываемого продукта от одного рабочего шага к другому.</p>
	<p>Сотрудники, команды или отделы, выполняющие действия в рамках рабочего шага.</p>
	<p>Данные о рабочем шаге. Содержат показатели и их значения, необходимые для понимания, как функционирует система; например, время выполнения (LT) = 22 часа, а время обработки (PT) = 1 час.</p> <p>Определения LT и PT приводятся в п. 2.2.1 «Метрики для анализа потока создания ценности».</p>
	<p>Запасы между рабочими шагами, т.е. число под символом, указывают на количество накапливающихся задач, в данном случае составляющее 30.</p> <p>Определение запасов приводится в п. 2.2.2 «Определение видов деятельности, не способствующих созданию ценности (избыточных)».</p>


	<p>Период выполнения каждого рабочего шага, включающий, как правило, время ожидания и обработки.</p>		
<table border="1" style="width: 100%;"> <tr> <td style="padding: 5px;">Общее время выполнения (LT)</td> </tr> <tr> <td style="padding: 5px;">Общее время обработки (PT)</td> </tr> </table>	Общее время выполнения (LT)	Общее время обработки (PT)	<p>Совокупность рабочих шагов для всего потока создания ценности, например, общее LT и общее PT.</p>
Общее время выполнения (LT)			
Общее время обработки (PT)			

Рисунок 2.3 Простые условные обозначения для визуализации потока создания ценности

Поскольку данная концепция позаимствована из сферы производства, доступных обозначений гораздо больше, особенно используемых для визуализации потока материалов и данных.

Дополнительные обозначения можно добавить в зависимости от контекста оптимизации после первой визуализации текущего статуса потоков создания ценности. Например, для более глубокого понимания формальных и неформальных информационных потоков визуализацию потока создания ценности можно проводить в комплексе с дополнительной визуализацией. В рамках тематического исследования, описанного в работе “FLOW-assisted value stream mapping in the early phases of large-scale software development” (Bin Ali et al., 2015), были выявлены проблемы первичного проведения визуализации текущего статуса потоков создания ценности. Для решения некоторых задач использовался метод дополнительного моделирования информационных потоков (FLOW).

Поскольку визуализация потока создания ценности используется в различных отраслях промышленности, рабочие шаги и содержание каждого шага могут различаться. Ниже приведено высокоуровневое описание стандартных рабочих шагов визуализация потока создания ценности:

1. Выяснить, на каком потоке сосредоточиться: в сфере операционной деятельности или связанным с разработкой
2. Определить начальную и конечную точки потока создания ценности, а также группы продуктов или услуг, для которых проводится визуализация
3. Осуществить визуализацию потока создания ценности для текущего статуса («как есть»), начиная либо с начала, либо с конца потока
4. Добавить ключевые показатели эффективности для каждого шага и определить «узкие места», задержки, проблемы с качеством и шаги, не связанные с созданием ценности (подробнее см. п. 2.2 «Анализ потока создания ценности с точки зрения качества и тестирования»)
5. Осуществить визуализацию потока создания ценности для будущего статуса с учетом изменений шагов и показателей эффективности
6. Согласовать и спланировать инициативы по оптимизации потоков создания ценности в отношении «узких мест», задержек, проблем с качеством и шагов, не связанных с созданием ценности

Текущий статус («как есть») можно визуализировать согласно диаграмме на рисунке 2.4 (метрики описаны в п. 2.2.1 «Метрики для анализа потока создания ценности»).

После первой визуализации потока создания ценности статус подлежит проверке и контролю на регулярной основе. По достижении показателей будущего статуса или по истечении определенного времени метод можно использовать повторно. В качестве альтернативы, метод можно использовать для визуализации других потоков создания ценности в организации или других продуктов или групп услуг в рамках того же потока создания ценности. При визуализации и анализе потоков создания ценности, итеративность — ключевая составляющая. Таким образом можно визуализировать данные текущего и будущего статуса для обеспечения непрерывной оптимизации потока создания ценности.

С точки зрения качества и тестирования метод визуализации потока создания ценности можно использовать для оптимизации процедуры тестирования и мер по обеспечению качества в более широкой перспективе, чем для одной команды, работающей по гибким методологиям.

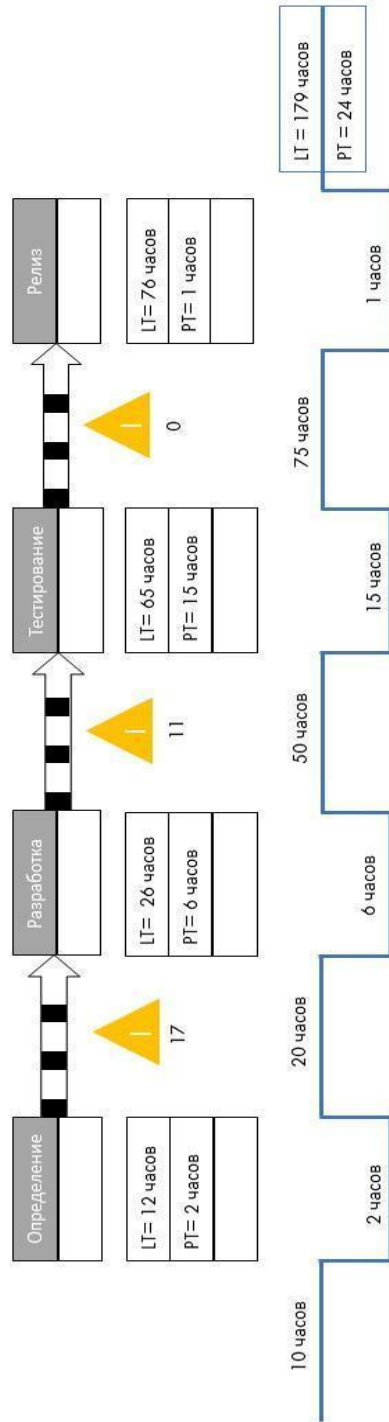


Рисунок 2.4 Базовая диаграмма «как есть» для потока создания ценности в сфере разработки

Метод лучше всего работает, если используется в небольшой группе, состоящей из сотрудников, осознающих значение различных рабочих шагов в потоке создания ценности, и включающей лидеров, которым следует способствовать выделению соответствующего бюджета и определению приоритетности для мер по оптимизации (Liker and Meier, 2005).

В контексте качества и тестирования метод визуализации потока создания ценности можно использовать как часть цикла непрерывной оптимизации (см. Главу 3 «Непрерывная оптимизация качества и тестирования»). Он также часто используется в организациях, чтобы понять, каким образом организовать поток создания ценности так, чтобы избежать разрозненности процессов. Это можно сделать в ходе командной ретроспективы для непрерывной оптимизации или семинара по визуализации потока создания ценности, поставив на повестку дня периодические ретроспективы. Важно, чтобы при принятии решения о том, как собрать сотрудников в команды, учитывалась и перспектива качества и тестирования.

Поскольку визуализация потока создания ценности предполагает более высокий уровень абстракции, чем отдельный процесс, данный метод не следует использовать для детального анализа процессов. Аналогичным образом, визуализация потока создания ценности требует широкой перспективы и не должна использоваться одним человеком или небольшой группой, которая включает сотрудников, выполняющих только одну функцию или один рабочий шаг в рамках потока создания ценности.

Если визуализация потока создания ценности еще не использовалась в организации (например, при содействии скрам-мастера, лидера, тренера по гибким методологиям или фасилитатора другого типа), метод может вызвать противодействие. Поскольку он требует участия разных сотрудников, важно заручиться поддержкой этих сотрудников и, возможно, их лидеров.

Получение и последовательное использование показателей эффективности на протяжении всего потока создания ценности может оказаться сложной задачей. Стандартные метрики, методы измерения и способы их использования при анализе потока создания ценности рассматриваются в следующем разделе.

2.2 Анализ потока создания ценности с точки зрения качества и тестирования

Меры по обеспечению качества и тестирования позволяют выявить недоработки в рамках каждого рабочего шага в процессе разработки продукта. Как правило, тестирование заключается в проверке качества функциональных и нефункциональных требований в начале поставки продукта и ближе к концу при проверке, в какой степени система соответствует заявленным требованиям и удовлетворяет потребности заказчика. В рамках масштабирования тестирования в организации с гибкими методологиями, помощь в обеспечении качества рассматривается как важная составляющая общей ответственности команд за качество, а лидеры гибкого тестирования и лидеры команд тестирования, работающих по гибким методологиям, должны проводить проверку качества процессов совместно с сотрудниками, которые внесли вклад в потоки создания ценности.

Визуализация потока создания ценности имеет много преимуществ, описанных в предыдущем разделе. Однако, чтобы понять, где скрываются проблемы или возможности оптимизации, важно измерить и проанализировать эффективность потока создания ценности. Эта мера требует итеративности.

Оптимизация потока создания ценности концентрируется на потоке ценности и на качестве. Таким образом, анализ потока создания ценности может стать мощным инструментом для всех тех, кто

применяет подход, основанный на принципе помощи в обеспечении качества, в отношении качества и тестирования. Такой анализ требует знания полной картины происходящего. Следовательно, лидеры гибкого тестирования и лидеры команд тестирования, работающих по гибким методологиям, могут помочь остальным изучить проблемы качества и тестирования в более широкой перспективе. Разумеется, также важно выявлять виды деятельности, способствующие созданию ценности, и продолжать их успешно осуществлять.

2.2.1 Метрики для анализа потока создания ценности

Организации стремятся, чтобы их продукция выходила на рынок в ожидаемом темпе и обладала ожидаемым качеством, соответствующим требованиям клиентов. Чтобы этого достичь, требуется четкое понимание характеристик потока производства на всех уровнях.

Для анализа потока создания ценности важно собирать данные о каждом рабочем шаге. Задача состоит в том, чтобы найти участки, на которых возможно повысить как результативность, так и эффективность потока создания ценности. Однако нельзя не подчеркнуть, насколько важно избегать локальной оптимизации, приводящей лишь к частичной оптимизации всего потока создания ценности. Таким образом, цель заключается в совершенствовании эффективности и результативности предоставления ценности клиенту в рамках потока создания ценности, что зачастую требует оптимизации управления качеством и связанных с тестированием мер.

Следующие метрики являются стандартными для анализа потока через поток создания ценности при разработке программного обеспечения:

- Время обработки (PT) (иногда также именуется «время контакта») — время работы над всеми задачами в рабочем шаге. Это время работы над продуктом и создания его ценности.
- Время ожидания (WT) (иногда также именуется «время переключения») — время, которое задача проводит в очередях между рабочими шагами, а не непосредственно в работе. Иногда время ожидания между задачами или действиями появляется даже в рамках рабочего шага, например, если владелец продукта не может в срок дать разъяснения для выполнения задачи.
- Время выполнения (LT) - интервал времени с начала деятельности в рамках рабочего шага до ее завершения, когда продукт уже подготовлен к следующему рабочему шагу. Другими словами, это сумма времени ожидания до начала рабочего шага и времени обработки в рамках рабочего шага.
- Эффективность потока (*flow efficiency*) (иногда также именуется «эффективность производственного цикла» или «коэффициент активности») — соотношение между общим временем обработки и общим временем выполнения для потока создания ценности.

$$\text{Эффективность потока (flow efficiency)} = \frac{PT_1 + PT_2 + \dots + PT_n}{LT_1 + LT_2 + \dots + LT_n} \times 100$$

Время обработки, время ожидания и время выполнения можно как измерять для одного рабочего шага, так и распространить для всего потока создания ценности.

Стандартные метрики для анализа качества:

- Процент завершенности и точности (%C&A) - Процент работы, которую на следующем рабочем шаге можно выполнить без необходимости внесения переделок и дополнительного запроса информации.
- Штучный процент завершенности и точности (*Rolled %C&A*) (также именуется «штучный сквозной выход») показывает вероятность того, что рабочий элемент пройдет через весь поток создания ценности без корректировок или поиска дополнительной информации.

$$Rolled \%C\&A = \%C\&A_1 \times \%C\&A_2 \times \dots \times \%C\&A_n \times 100$$

где %C&A₁ означает процент завершенности и точности для рабочего шага 1, %C&A₂ означает процент завершенности и точности для рабочего шага 2, а %C&A_n означает процент завершенности и точности для рабочего шага n.

- Эффективность фазовой локализации (*PCE*) - процент дефектов¹, которые были внесены, найдены и исправлены на одном и том же рабочем шаге разработки программного обеспечения, по отношению к дефектам, внесенным на этом рабочем шаге и найденным на этом и последующих рабочих шагах. Данная метрика отличается от процента обнаружения дефектов (DDP), поскольку сосредоточена не на этапе тестирования (уровень тестирования), а на рабочем шаге в потоке создания ценности, и включает только дефекты, допущенные в рамках рабочего шага, для которого измеряется *PCE*.

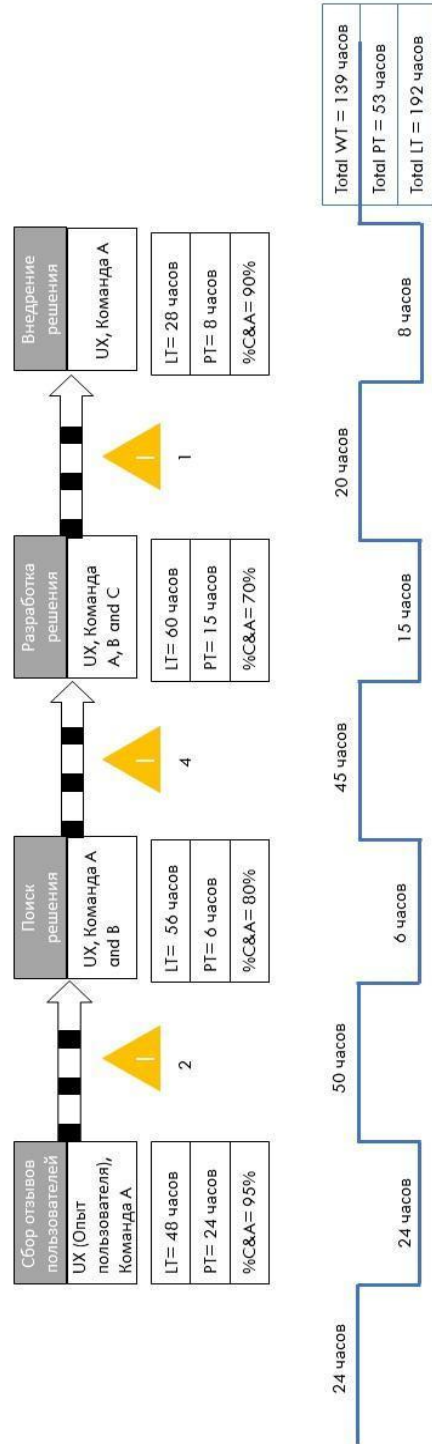
$$PCE = \frac{Df_1}{Df_1 + Df_{1a}} \times 100$$

где *Df*₁ означает дефекты, допущенные и выявленные в рамках рабочего шага 1, а *Df*_{1a} - дефекты, допущенные в рамках рабочего шага 1 и выявленные на последующих рабочих шагах.

Схема на **рисунке 2.5** представляет собой пример визуализации потока создания ценности с добавлением основных измерений для каждого рабочего шага.

Метрики необходимы для анализа потока создания ценности, однако последовательное проведение измерений на протяжении потока создания ценности может оказаться непростой задачей. В качестве отправной точки можно использовать уже имеющиеся данные. Если такие данные отсутствуют, группе, занимающейся визуализацией потока создания ценности, необходимо будет найти сотрудников, которые помогут провести оценку еще не измеренных и не собранных данных.

¹ Определения ошибки, дефекта и отказов, используемые ISTQB®, отличаются от определений, приведенных в общей литературе по бережливому производству, например, Lean Lexicon (Lean Enterprise Institute, 2014). В настоящем документе термин «дефект» используется в соответствии с глоссарием ISTQB®.



Эффективность потока (Flow Efficiency) $(53 / 192 * 100) = 28\%$
 Штучный процент завершения и точности %C&A $(0.95 * 0.8 * 0.7 * 0.9 * 100) = 48\%$

Рисунок 2.5 Базовая диаграмма текущего статуса с измерениями потока и качества

Группе следует буквально «выйти в поле», чтобы проверить, как работают сотрудники в рамках всего потока создания ценности. Этот принцип также именуется «Генчи Генбуцу» (Wikipedia, 2019). Наблюдая за сотрудниками, командами и отделами, занятыми в рамках потока создания ценности, и общаясь с ними, группа, проводящая анализ потока создания ценности, способна:

- Понимать содержание рабочих шагов и их связь друг с другом
- Обсуждать как данные, собранные работниками в рамках каждого рабочего шага, так и необходимость измерений
- Отслеживать деятельность, не способствующую созданию ценности (потери)
- Определить причины, лежащие в основе деятельности, не способствующей созданию ценности
- Сотрудничать в части оптимизации

Показатели качества, такие как %C&A и PCE, полезны для выявления проблем качества в рамках рабочего шага. Руководствуясь примером на **рисунке 2.5**, группа может обсудить, почему %C&A как поиска, так и разработки решения ниже, чем %C&A получения представлений пользователя и внедрения решения, и как это влияет на время выполнения и общий поток. В примере на **рисунке 2.6** обсуждение можно сосредоточить на высоком проценте дефектов, выявленных на последующих рабочих шагах. Оно может включать в себя беседу о том, как принять меры по обеспечению и контролю качества.

Аналогичным образом, для выявления «узких мест» можно проанализировать причины продолжительного времени ожидания, снижающего эффективность потока. Наличие проблемных областей в этом случае может быть напрямую или косвенно связано с качеством и тестированием. Например, если тестирование проводится вручную и хаотично, то очередь из элементов, подлежащих тестированию, постоянно растет.

Как и всегда, в работе с метриками следует уделить особое внимание тому, чтобы все сотрудники понимали:

- Назначение выбранных метрик
- Как следует использовать метрики и как избежать неверного их использования
- Кто должен проводить измерения и как они проводятся

Некоторые метрики, используемые для анализа потока создания ценности, можно применять только в течение ограниченного периода времени, чтобы изучить конкретные проблемы и измерить результаты совершенствований. Пример такой метрики — Эффективность фазовой локализации (PCE).

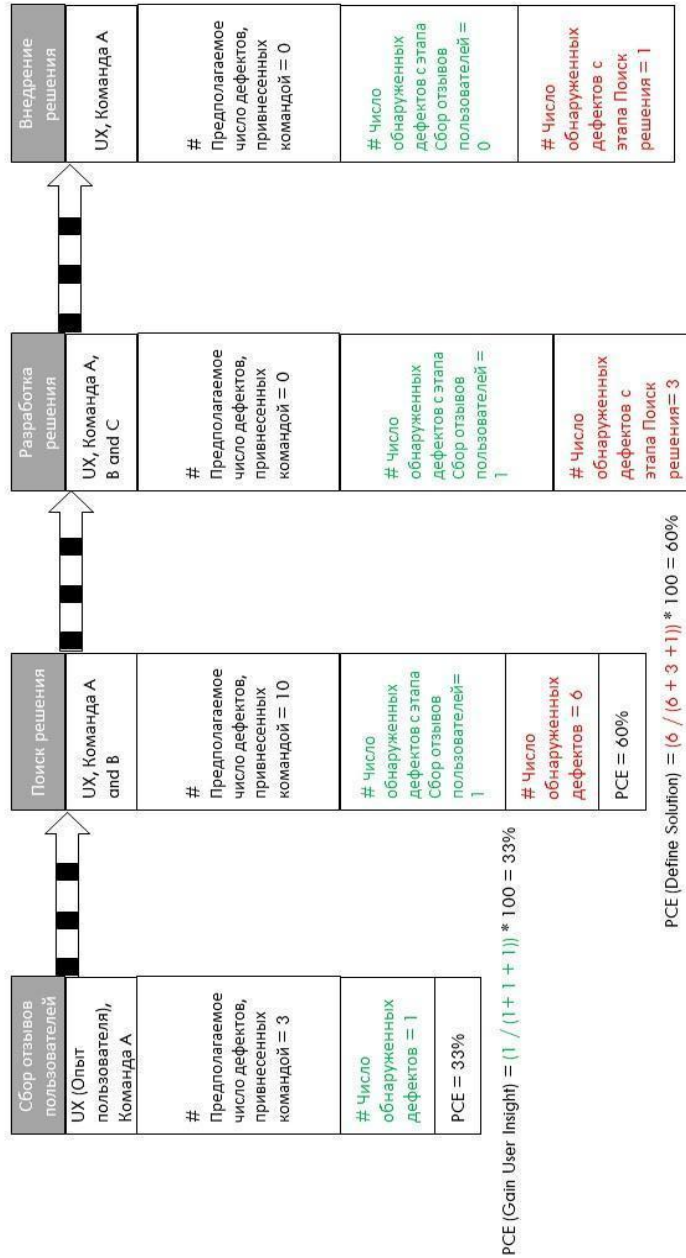


Рисунок 2.6 Пример эффективности фазовой локализации

2.2.2 Определение видов деятельности, не способствующих созданию ценности (Потери)

Существует несколько способов выявления в процессе обеспечения качества и тестирования видов деятельности, не способствующих созданию ценности.

Ниже приведены восемь примеров потерь.

- **Транспортировка** - перемещение незавершенного производства с места на место в процессе работы (Liker and Meier, 2005). Сюда относится обмен продуктами, информацией и материалами: например, если несколько удаленных тестировщиков обмениваются слишком большим количеством информации по электронной почте в дополнение к собраниям команды, на которых все они присутствуют. Чрезмерный обмен информацией может привести к ошибкам и корректировкам.
- **Запасы** - количество запасов больше минимально необходимого (Lean Enterprise Institute, 2014). Сюда относятся элементы, ожидающие внедрения в процесс для последующей разработки, или простаивающие в очереди, потому что никто с ними не работает, например, когда тестировщики создают подробные тесты для будущего использования, но откладывают внедрение важных решений по архитектуре системы. Ожидается, что такие решения не будут приниматься в краткосрочной перспективе, поэтому тесты уходят в запас и могут потребовать дополнительной работы после принятия решений.
- **Перемещение** - ненужные меры или действия на рабочем шаге или между рабочими шагами, которые не способствуют созданию ценности продукта (Lean Enterprise Institute, 2014), например, необходимость вносить изменения в статус отчета о дефектах, потому что рабочий процесс в инструменте управления дефектами не позволяет пропустить этапы, даже если это не помогает координировать работу в течение жизненного цикла дефекта.
- **Ожидание** - бездействие операторов (Lean Enterprise Institute, 2014). Нахождение любого сотрудника в ожидании (информации, работы, выполняемой другими, доступа к машине или ресурсу), например, когда тестировщики не могут выполнять работу из-за снижения скорости доступа в сеть или в случаях, когда из-за сбоя тестовой среды прерывается выполнение теста.
- **Перепроизводство**: производство элементов, потери для последующего процесса или для клиента (Lean Enterprise Institute, 2014), например, когда менеджер по тестированию создает объемные планы тестов и отчеты о тестировании, которые никто не читает или которые не являются действующими документами.
- **Излишняя обработка**: ненужная или неверная обработка (Lean Enterprise Institute, 2014). Переизбыток действий в рамках рабочего шага или ненужные рабочие шаги: например, когда перед запуском новой функции релиз должны утвердить много разных служб в одной компании. В некоторых случаях работа таких служб — простая формальность.

- **Коррекция: инспекция², доработка и брак** (Lean Enterprise Institute, 2014). Обратите внимание, что проверка в рамках бережливого производства может включать в себя поздние этапы системного тестирования, тогда как этого можно избежать. Брак включает в себя дефекты, прошедшие через поток создания ценности, что приводит к переделкам, и, например, случаи, когда лидер команды тестирования, работающей по гибким методологиям, обнаруживает, что конфигурация тестовой среды всегда требует ряда циклов корректировки.
- **Неиспользованный человеческий потенциал:** неиспользование отзывов сотрудников для совершенствования процесса и отсутствие возможностей развития сотрудников (Brito et al., 2020). Сюда относится отказ от поддержки сотрудников в их развитии, приобретении новых навыков и компетенций, например, неиспользование навыков, опыта и знаний сотрудников при назначении на определенные должности.

В ходе анализа и оптимизации потоков создания ценности в организации лидерам гибкого тестирования и лидерам команды тестирования, работающей по гибким методологиям, следует руководствоваться принципом бережливого мышления. Визуализация потока создания ценности помогает идентифицировать избыточные процессы в рамках как операционной деятельности, так и потока создания ценности в сфере разработки. В ситуации низкой эффективности или неэффективности существует несколько типовых стратегий выявления избыточных процессов в потоке создания ценности:

- **Поиск рабочих элементов, накапливающихся до и после каждого рабочего шага.** Такое, как правило, случается при ожидании участниками команды своей очереди. Например, недостаток оповещений (отсутствие визуальных сигналов) о том, что рабочие элементы готовы к следующему рабочему шагу, и о том, как передается информация, приводит к неэффективной передаче очереди. Следовательно, сокращение и даже устранение этих проблем поможет сократить время выполнения.
- **Отслеживание на каждом рабочем шаге результата работы и сотрудников, ее выполняющих,** что позволит сократить время обработки. Такой подход позволит выявить и обратное, например, когда откладываются важные меры по тестированию или обеспечению качества, что приводит к накоплению задач по качеству и снижению метрики эффективности фазовой локализации (PCE).
- **Поиск и количественная оценка дефектов до и после каждого рабочего шага.** Большое количество дефектов говорит о наличии потерь. Если время обработки растет, а количество дефектов остается неизменным, это указывает на то, что в рамках рабочего шага не выявлены некоторые дефекты или накоплены технические задачи. Таким образом, количественная оценка дефектов помогает выявить возможности для внедрения

² Определение инспекция, используемые ISTQB®, отличаются от определений, приведенных в общей литературе по бережливому производству, например, Lean Lexicon (Lean Enterprise Institute, 2014).

встроенных мер по обеспечению качества, особенно для потоков создания ценности в сфере разработки.

- Оценка количества запросов на поддержку от клиентов или других заинтересованных сторон, что может быть вызвано проблемами с качеством. Обработка таких запросов способна прервать работу по доставке продукта и негативно сказаться на времени выполнения и времени обработки.

Схема на **рисунке 2.7** содержит пример визуализации будущего статуса, предполагающего выявление всех проблем. Группа определила некоторые целевые показатели для повышения производительности.

Визуализация будущего статуса является целью, но не углубленным анализом в поиске решения, как достичь будущего статуса со всеми инициативами по оптимизации. Основная задача визуализации заключается в выявлении критических точек в потоке создания ценности и получении представления о том, как использовать их для создания ценности, в частности, совершенствования качества и сокращения времени выполнения при меньших затратах. Порядок определения, планирования и проведения этапов оптимизации с использованием структурированного подхода к решению проблем описан в **главе 3**.

Анализ и совершенствование потоков создания ценности в основном зависит от получения знаний, позволяющих отслеживать рабочие потоки и дающих сотрудникам возможность по-новому организовать обеспечение качества. Соответственно, лидерам гибкого тестирования следует способствовать данному процессу несколькими способами, например:

- Продвигать комплексный подход к анализу проблем и оптимизировать потоки создания ценности
- Помогать сотрудникам в профессиональном развитии и понимать, как качество и тестирование влияют на эффективность потока создания ценности
- Способствовать внедрению принципа «встроенного качества» и обучить ему сотрудников, например:
 - Поощрять получение глубоких знаний о продукте создающими его сотрудниками: это позволяет обнаруживать дефекты до того, как их обнаружат клиенты, при этом сокращая время выполнения
 - Продвигать и поддерживать внедрение разработки через тестирование
 - Продвигать культуру немедленного исправления дефектов, чтобы обеспечить непрерывное создание ценности для клиента взамен активного тестирования на конечном этапе
 - Для устранения критических дефектов и купирования проблем может потребоваться внедрить принцип командного поведения. При наличии нескольких команд гибкой разработки и частых релизов он позволяет избежать потерь важной информации из-за быстрой смены обстоятельств. Кроме того, такой принцип предотвращает возникновение новых препятствий для создания ПО, ведущих к появлению дефектов
 - Провести анализ первопричин появления дефектов в рамках «сдвига влево»; это позволит изменить сам способ разработки и тестирования и упростить поставку продуктов.
- В контексте потока создания ценности в сфере операционной деятельности: способствовать выявлению проблем в обеспечении качества в рамках цикла взаимодействия с клиентом

- Поддерживать вовлечение клиентов или конечных пользователей в поток создания ценности, например, посредством:
 - Взаимодействия с бета-тестировщиками
 - Проведения регулярных практикумов по изучению пользовательских историй на базе разработки через приемочное тестирование (ATDD)
 - Сессии исследовательского тестирования с участием клиентов или конечных пользователей

Лидеры гибкого тестирования и лидеры команд тестирования, работающих по гибким методологиям, могут содействовать разработке инициатив по оптимизации в целях сокращения избыточных процессов посредством ряда циклов Деминга (PDCA), см. п. 3.1 **«Структурированный подход к решению проблем тестирования и мер по обеспечению качества»**.

Если организация осознает значение помощи в обеспечении качества, визуализация потока создания ценности может стать мощным инструментом, который необходимо внедрить в рамках мер помощи в обеспечении качества.

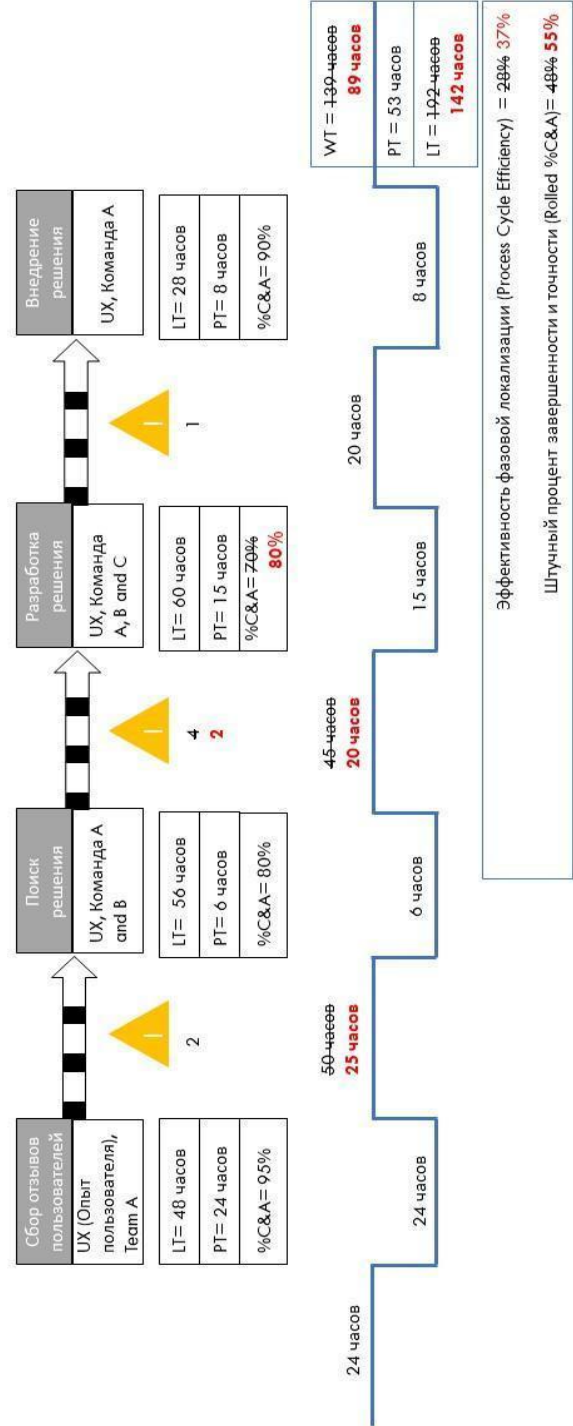


Рисунок 2.7 Пример визуализации будущего статуса; цели оптимизации выделены красным цветом

3 Непрерывное совершенствование качества и тестирования – 150 минут

Как описано в главе 2 «Визуализация потока создания ценности», визуализация и анализ потока создания ценности помогают определить проблемы качества и тестирования в рамках рабочего шага потока создания ценности, которые могут быть вызваны событиями, происходящими на предыдущем шаге, или создавать дополнительные проблемы на последующем шаге. Один из примеров — не выявленные в процессе анализа потребности пользователей. Если не выявить потребности пользователя на ранних этапах, может оказаться так, что полностью готовый продукт не решает проблему пользователя. На первый взгляд, такой проблемы легко избежать. Однако понять, когда что-то идет не так в серии рабочих шагов, и, что не менее важно, установить основные причины, по которым что-то идет не так — сложная задача. По получении углубленного представления по этим вопросам можно проводить эксперименты по оптимизации, чтобы решить проблему и предотвратить ее повторение в будущем. Цикл Деминга (PDCA) — метод организации совершенствования путем эксперимента, разработанный Уильямом Эдвардсом Демингом.

Лидеры гибкого тестирования могут способствовать организационному обучению, внедряя структурированный подход к решению проблем. Кроме того, решение проблем, охватывающих несколько групп и, возможно, даже несколько потоков создания ценности, требует более масштабного и комплексного подхода. Диаграмма причинно-следственных связей — метод, полезный в этом контексте при анализе первопричин и в ходе ретроспективы в целом.

3.1 Структурированный подход к решению проблем в сфере тестирования и мер по обеспечению качества

Иногда организации, ориентированной на ценность для решения проблем, требуется участие нескольких гибких команд, а также анализ нескольких потоков создания ценности, как обсуждалось в главе 2. В таком случае требуется подход, учитывающий целостный взгляд, объединяющий практики бережливого производства и использования гибких методологий. Лидеры гибкого тестирования и лидеры команд тестирования, работающих по гибким методологиям, должны понимать и использовать практики системного мышления для определения первопричины в сложных условиях.

3.1.1 Цикл Деминга (PDCA)

Цикл Деминга PDCA (Plan-Do-Check-Act, планирование - выполнение - проверка - воздействие) — практический подход к решению проблем и обеспечению непрерывной оптимизации. Цикл Деминга (PDCA) можно применять в ходе экспериментов по локальной оптимизации, например, сокращения времени тестирования производительности веб-интерфейса. Циклы Деминга (PDCA) можно по аналогии применять и в рамках более масштабных инициатив по оптимизации, например, для уменьшения количества дефектов в работе нескольких взаимозависимых команд, работающих по гибким методологиям, занятых одними и теми же продуктами.

Обычно применение цикла Деминга (PDCA) начинается с анализа разрывов, в рамках которого сначала описывается цель, а затем — фактическая ситуация. В разнице между ними заключаются «разрывы». Они возникают, например, при достижении новой цели или повышении производительности в существующей системе. Затем разрыв можно устранить с помощью цикла Деминга (PDCA) путем ряда совершенствований в рамках рабочего шага.

Цикл Деминга (PDCA) тесно связан с управлением качеством. Использование цикла Деминга (PDCA) позволит организации эффективно контролировать и непрерывно оптимизировать результативность и качество своих поставок, а тем самым — и ценность для клиента. В условиях гибкой разработки в масштабе предприятия, когда обеспечение качества считается совместной ответственностью, цикл Деминга (PDCA) имеет следующие преимущества:

- Минимизация повторяющихся проблем с качеством.
- Поддержка сотрудников в формировании идей оптимизации. Это позволит совершенствовать работу всей организации, при этом продвигаясь к гибкости бизнеса.
- Систематическое и углубленное отслеживание всех потоков создания ценностей. В результате совершенствования понимание сотрудниками производительности и обеспечения качества в организации.
- При повторении циклов Деминга (PDCA) накапливается опыт, на котором можно учиться и сотрудникам, и организациям.

Основополагающий принцип цикла Деминга (PDCA) — повторяемость. Цикл Деминга (PDCA) начинается с этапа планирования, на котором осуществляется осознание проблемы или бизнес-возможности, формирование гипотезы и разработка плана реализации, включая проверку достоверности гипотезы с использованием метрик.

На этапе выполнения разработанный план претворяется в жизнь.

Целью этапа проверки является измерение влияния мер, проведенных на этапе выполнения. Проверка требует сравнения фактических результатов с целевыми прогнозами, определенными на этапе планирования. При отрицательных результатах проверки цикл Деминга (PDCA) зачастую приходится начинать заново. В свою очередь, результаты выше целевых дают возможности для новых циклов Деминга (PDCA).

На этапе воздействия на основе полученных результатов принимаются меры по устойчивому внедрению новых методов работы. Сюда можно отнести меры по стандартизации процессов.

Приведем пример того, как цикл Деминга (PDCA) можно использовать для решения проблем в нескольких командах тестирования, работающих по гибким методологиям:

Две команды разработчиков продукта для важного клиента получили отрицательный отзыв о функциональном качестве продукта.

- Планирование: Лидер гибкого тестирования рассматривает обратную связь как возможность идентифицировать и обсудить воспринимаемую проблему с владельцем продукта. Выдвигается гипотеза о том, что одной из первопричин проблемы является отсутствие конкретных и ориентированных на клиента требований к производительности, что приводит к недостаткам реализации и тестирования. Лидер тестирования по итогам беседы с двумя командами и доступными обеим командам контактными лицами клиента разрабатывает план: опробовать формат семинара, в котором участвовали бы как члены команд, так и представители клиента. Этап планирования цикла Деминга (PDCA) заключается в проведении семинара и обеспечении следующего:
 - Обе команды подтверждают, что формат существенно совершенствуют понимание ими сфер, важных для производительности клиента
 - Формат семинара позволяет определить пределы производительности для тестирования

- Клиент подтверждает, что участие в семинарах на постоянной основе для него приемлемо
- В ходе дальнейшей разработки полученные на семинаре результаты используются для тестирования, что, соответственно, дает релевантный результат
- **Выполнение:** по завершении семинара обе команды используют полученные на семинаре критерии эффективности при разработке и тестировании последующих итераций.
- **Проверка:** в ходе ретроспективы после релиза выполняются все критерии, установленные на этапе планирования. Дополнительная обратная связь от участников команд заключается в том, что уточняется отзыв клиента для некоторых видов тестирования производительности.
- **Воздействие:** описание формата семинара документируется в базе знаний, и владелец продукта соглашается проводить с клиентом по крайней мере один семинар по нефункциональным требованиям каждый квартал. Последнее решение можно рассматривать как этап планирования следующего цикла Деминга (PDCA). Кроме того, команды сотрудничают с владельцами и других продуктов для использования дополнительных отраслевых критериев в качестве критериев приемлемой производительности, которые затем принимаются за стандарт.

В следующем примере этап воздействия пропускается, потому что ожидаемые совершенствования не могут быть реализованы в рамках пилотного проекта:

Несколько команд, работающих по гибким методологиям, оспаривают целесообразность использования в компании инструмента управления тестированием. Инструмент выглядит устаревшим и не обновляется. Однако, несмотря на очевидные недостатки, официально от инструмента не отказываются. Некоторые команды уже подыскали инструмент на замену, позволяющий подключить тестирование к непрерывной интеграции (CI); некоторые утверждают, что существующий инструмент — единственный способ получить результаты системного тестирования для поддержки регрессионного тестирования. Три команды проводят групповую ретроспективу по инструменту управления тестированием и решают проверить, можно ли найти инструмент на замену, проверив, как каждая команда будет пользоваться разным инструментом.

- **План:** лидер гибкого тестирования проводит консультации со скрам-мастерами различных команд, проводится ретроспектива. Разрабатывается план с целью подобрать инструмент управления тестированием, который стал бы обязательным для использования всеми командами, работающими по гибким методологиям. Один из пилотных проектов, основанный на технологии, которую не смогут использовать другие команды, решено остановить, при этом продолжив работу над двумя другими проектами. Также решено, что оба проекта, участвующие в эксперименте, будут проводить взаимные ревью.
- **Выполнение:** на этом этапе обе команды продолжают работу над пилотным проектом, фиксируя положительный и отрицательный опыт использования соответствующих инструментов. Обе команды приходят к выводу, что хотели бы продолжить работу со своим инструментом.
- **Проверка:** каждая команда рассматривает полученные результаты и выводы, сделанные другой командой. Однако команда, которая сама не опробовала инструмент, не приходит к такому же положительному выводу, как и другая команда. Результаты обсуждаются на рабочем совещании, с тем чтобы определить, может ли одновременное использование двух инструментов стать долгосрочным решением. Поскольку суть договоренности была в том, что следует выбрать один общий инструмент, команды не переходят к этапу корректировки, а соглашаются, что необходим новый план.

В обеспечении качества лидер гибкого тестирования может упростить цикл Деминга (PDCA) для разрешения большого количества «незавершенных и неточных» рабочих шагов в потоке создания ценности. При визуализации будущего статуса потоков создания ценности команды могут проверить некоторые допущения и свои действия в контексте различных циклов Деминга (PDCA): например, большое количество «незавершенных и неточных» рабочих шагов может быть одной из основных причин слишком продолжительного времени выполнения.

Подробнее о дальнейшем использовании различных методов совершенствования процесса тестирования см. программу обучения “Программа подготовки ISTQB Экспертного уровня Улучшение тестового процесса” (ISTQB®, 2011).

3.1.2 Внедрение цикла Деминга (PDCA) в организации

Как упоминалось выше, метод PDCA можно использовать для проведения активностей локальной оптимизации и более масштабных инициатив по оптимизации.

В контексте гибкой разработки в масштабе предприятия типовыми организационными условиями для внедрения цикла Деминга (PDCA) в процессы разработки и тестирования ПО являются:

- Ретроспектива с участием нескольких команд
- Команды по оптимизации проектов и программ
- Время для обсуждения мер по оптимизации в ходе совещаний по планированию релиза

Для внедрения цикла Деминга (PDCA) в контексте гибкого бизнеса требуется:

- Общее понимание проблемы перед началом цикла Деминга (PDCA). Проблема — это несоответствие ожидаемых и реальных показателей. Неожиданно большое количество дефектов, дополнительное время выполнения (LT) при релизе, недовольный клиент — все это проблемы. Проблемы производительности должны выражаться в количественных показателях. Поэтому цикл Деминга (PDCA) следует стартовать, когда станут известны как текущий (например, 8) так и целевой (например, 9) показатели удовлетворенности.
- Корректность описания проблемы имеет решающее значение. Например, представитель клиента может воспринимать сдвиг сроков как разрыв во времени выполнения (LT): «Мы потратили 23 дня вместо запланированных 10». Всем сотрудникам следует научиться точно отчитываться о проделанной работе, установив такие правила на корпоративном уровне. Этому способствует подход, основанный на помощи в обеспечении качества.
- Способность выявлять проблемы и возможности в масштабе организации. Каждый сотрудник должен иметь возможность выявлять проблемы на любом этапе и решать их надлежащим образом по мере возникновения. Для этого необходима обстановка, стимулирующая выявление недочетов. Культура умения ошибаться, в рамках которой неудача рассматривается как возможность учиться, имеет важное значение для оптимизации на организационном уровне.

Проблема или возможность, для которой целесообразно провести цикл Деминга (PDCA), должна напрямую влиять на удовлетворенность клиентов. Решение такой проблемы совершенствует способность организации приносить клиентам пользу.

Команды тестирования, работающие по гибким методологиям, могут проводить циклы Деминга (PDCA) в ходе нескольких ретроспектив. Например, во время первой ретроспективы команда согласовывает и определяет проблему, проводит наблюдение и разрабатывает план. Затем, во

время спринта, команда выполняет запланированные действия. Таким образом, следующую ретроспективу команды в рамках цикла Деминга (PDCA) можно посвятить этапу проверки.

Этап воздействия предполагает извлечение выводов из действий, предусмотренных и осуществленных в ходе выполнения плана. Прежде всего, этот шаг касается того, что требуется изменить в рабочем процессе. В результате, как правило, создаются рабочие стандарты, которые используются до тех пор, пока в ходе следующего цикла Деминга (PDCA) не определятся более подходящие меры (Liker and Meier, 2005). Для внедрения рабочего стандарта на предприятии необходимо использовать теоретические и практические тренинги в паре.

Внедрение цикла Деминга (PDCA) в организации часто требует обучения. Возможности обучения могут возникнуть в процессе повседневной работы, или же их можно активно искать, сравнивая текущую ситуацию с опытом других команд, компаний или моделей, пытающихся измерить зрелость гибких технологий и процессов.

Многие гибкие системы масштабирования предполагают измерение и, следовательно, повышение прозрачности организационной эффективности предприятия. В качестве основы требуется безопасная прозрачная обстановка — в противном случае измерение зрелости команд и потока создания ценности может подтолкнуть отдельных сотрудников и команды к сокрытию истинного положения дел для того, чтобы не прослыть неэффективными. С другой стороны, если компания разработала подходящую модель зрелости, она может помочь проведению циклов Деминга (PDCA).

Команды, принимающие меры по локальной оптимизации на этапе выполнения, также должны завершить следующие этапы:

- Идентификация — например:
 - Определить метрики тестирования, которые могут использовать другие отделы в организации
 - Проанализировать ответы от системы заявок пользовательской справочной службы и зафиксировать результат в системе управления конфигурацией, чтобы он был виден всем
 - Провести детальный анализ, какого рода измерения производятся в конвейере CI, и создать соответствующее описание во внутренней системе управления знаниями
- Выравнивание — например:
 - Активно способствовать обсуждению с разработчиками ПО вопроса о возможности тестирования
 - Передать профессиональному сообществу сведения о проблеме и побудить их внести вклад в меры по оптимизации
- Реализация — например:
 - Создать прозрачную (в достаточной степени) документацию в рамках проведения активностей по оптимизации
- Продемонстрировать локальные активности, выявленные в рамках цикла Деминга (PDCA) всей организации, чтобы позволить внедрять совершенствования при необходимости и проводить организационное обучение
 - Передать профессиональному сообществу результаты активностей
 - Использовать большие заметные мониторы для демонстрации конвейера CI
 - Убедиться, что активности по оптимизации и их результаты доступны в системах управления конфигурацией за пределами команды

- Включить результаты командных мер по оптимизации в обсуждениях в ходе ретроспективы с участием нескольких команд

Идентификация, выравнивание, реализация и демонстрация могут считаться вторым циклом в рамках этапа выполнения (Bendek, 2018).

Подводя итоги, можно сказать, что условия для поддержания культуры цикла Деминга (PDCA) в масштабе предприятия могут иметь следующее значение:

- Команды действуют в целях раскрытия потенциала оптимизации на организационном уровне.
- В рамках организации поощряются официальные совещания и процессы, направленные на оптимизацию работы выше уровня команды.
- Руководство поддерживает связанные с циклом Деминга (PDCA) межпроектные совещания и процедуры, поскольку без этого большинство локальных активностей по оптимизации не принесут масштабного эффекта.

3.2 Системное мышление и анализ первопричин

Руководство помощью в обеспечении качества на организационном и стратегическом уровнях требует более масштабного подхода, чем при организации одного мероприятия, проекта или отдела. Системное мышление и анализ первопричин являются важными дисциплинами, предоставляющими множество различных методов анализа сложных проблем. Лидерам гибкого тестирования следует участвовать в анализе сложных проблем и упрощать его, чтобы помочь организации развиваться и оптимизировать потоки создания ценности.

3.2.1 Системное мышление

Системное мышление является важнейшей дисциплиной масштабирования гибкости при переходе от метода разработки ПО, используемого в первую очередь ИТ-командами, к методу создания ценности, в котором задействована вся организация. В некоторых фреймворках для масштабирования гибкости системное мышление считается одним из ключевых принципов, например, в LeSS (The LeSS Company B.V., без даты) и SAFe (Scaled Agile Inc., 2019).

Существует множество определений системного мышления, но все их объединяют некоторые общие характеристики. Следующий их список приводится в работе Stave and Hopper (2007):

- Признание взаимосвязей: видение всей системы и понимание того, как части системы соотносятся с целым.
- Выявление обратной связи: выявление причинно-следственных связей между частями системы, описание цепочек причинно-следственных связей, признание существования обратной связи в замкнутых цепочках причинно-следственных связей, выявление разнообразия отношений и циклов обратной связи.
- Понимание принципа динамического поведения: понимание того, что обратная связь влечет за собой создание моделей поведения, демонстрируемых системой, определение системных проблем с точки зрения динамического поведения, восприятие поведения системы как функции внутренней структуры, а не реакции на внешний стимул, понимание типов моделей поведения, связанных с различными типами структур обратной связи, признание влияния задержек на поведение.

- Дифференциация типов потоков и переменных: понимание разницы между способностью идентифицировать потоки: ставок, уровней, материалов и информации, а также понимание того, как различные переменные работают в составе системы.
- Использование концептуальных моделей: синтез и применение концепций причинно-следственной связи, обратной связи и типов переменных.
- Создание моделей: описания системных соединений в математических терминах.³
- Регламенты тестирования: использование моделей для определения точек воздействия и проверки гипотез для принятия решений.

В системном мышлении поток создания ценности — является одним из типов систем, существующих в организации (Scaled Agile Inc., 2019). Важно, чтобы поток создания ценности был оптимизирован полностью, а не частично. Организация также является своего рода системой, такой же, как и технические системы (Scaled Agile Inc., 2019). Системное мышление включает в себя выявление и понимание систем, прогнозирование их поведения и разработку модификаций для получения желаемого эффекта (Arnold and Wade, 2007).

Питер Сенге считает системное мышление пятой дисциплиной, необходимой для создания обучающейся организации. Одна из проблем, связанных с обучением, заключается в том, что причина и следствие не всегда тесно связаны во времени и пространстве. Наиболее важные решения, принимаемые во многих организациях, имеют общесистемные последствия, которые можно наблюдать в течение нескольких лет (Senge, 1990).

Методы системного мышления помогают, в частности (LeSS Company B.V., дата не указана):

- Понимать динамику системы
- Выявлять первопричины процессов в сложных системах
- Ревизовать существующие модели мышления
- Избежать локальной оптимизации

Как упоминалось выше в **главах 1 и 2**, при внедрении помощи в обеспечении качества лидеры гибкого тестирования помогают оптимизировать всю систему, а не только команду или отдел тестирования. Если лидеры гибкого тестирования не обладают базовыми навыками системного мышления, существует риск локальной оптимизации (т.е. изменений, которые улучшат тестирование, но приведут к снижению общей ценности). Риски возрастают при использовании традиционного подхода к разработке, поскольку меры по качеству и тестированию часто рассматриваются как часть конкретного этапа, особенно если этап тестирования следует после завершения этапа разработки.

Системное мышление можно использовать во множестве ситуаций, например, при устранении проблем, принятии решений, проведении ретроспектив с участием нескольких команд и оптимизации процессов. Далее рассматриваются два метода, используемые в системном мышлении.

3.2.2 Анализ первопричин

Как упоминалось в программе обучения базового уровня «Сертифицированный тестировщик» (ISTQB®, 2018), анализ первопричин — основа успешной ретроспективы

³ Некоторые исследователи считают моделирование продвинутым компонентом системного мышления. Другие рассматривают его как понятие, выходящее за рамки определения системного мышления (Stave and Hopper, 2007).

Если внедрение системы или решения требует сотрудничества команд, работающих по гибким методологиям, некоторые меры обеспечения качества и тестирования охватывают несколько команд и ответственность за рабочие решения, как правило, распределяется между ними. При возникновении проблем команды должны взаимодействовать, чтобы понять, что их вызывает и как их эффективно решать. Если одна команда пытается устранить проблему, не имея полного представления о решении, это может повлечь за собой новые проблемы для других команд или возыметь действие лишь в ограниченном количестве ситуаций.

К потенциальным проблемам, которые часто затрагивают несколько команд, относятся:

- Отказы после выпуска в эксплуатацию
- Нестабильные тестовые среды
- Некачественное автотестирование
- Проектирование с обеспечением тестопригодности (чрезвычайно важный аспект системы)
- Проблемы с интеграцией систем или решений, предоставляемых внешними партнерами или поставщиками
- Определенные уровни/типы тестирования (например, системное тестирование, системное интеграционное тестирование, интеграционное тестирование аппаратного и программного обеспечения, тестирование системы систем и тестирование в условиях эксплуатации)
- Методы проверки бизнес-гипотез

Помимо проблем, есть и другие веские причины сосредоточиться на обеспечении качества и тестировании при участии в разработке нескольких команд:

- Снижение затрат на лицензирование или других затрат, связанных с инструментами, оборудованием и прочими элементами, необходимыми для обеспечения качества и тестирования
- Выявление и использование эффектов синергии при совместном использовании инструментов и процессов
- Общее совершенствование и оптимизация мер по обеспечению качества и проведению тестирования
- Непрерывное внимание к важным сферам (высокие риски для бизнеса), чтобы сохранить текущее состояние и избежать возврата назад
- Обеспечение получения знаний и общего понимания системы и процессов.

Определение узких мест в потоке создания ценности часто означает обнаружение основной причины избыточных процессов. Теория ограничений (*Theory of constraints*, TOC), изложенная в *The Goal* (Goldratt and Cox, 2004), содержит практические советы о том, как искать узкие места в различных системах. При переходе организации от традиционной к гибкой разработке такие проблемные области могут возникнуть в потоке создания ценности в сфере разработки. Приводим примеры в порядке возрастания уровня зрелости процессов:

1. Организация сред: тестировщики могут предоставить методы автоматизированной настройки тестовых сред для всего потока создания ценности. Возможно, им потребуется ознакомиться с новыми технологическими трендами, например подходом

- «инфраструктура как код». Среда с сервисом самообслуживания может помочь избежать образования узких мест.
2. Развертывание кода: тестировщики могут помочь организовать безопасную и надежную автоматизацию развертывания кода.
 3. Системное тестирование: инициативой может быть массовая автоматизация и распараллеливание системных тестов. Другая инициатива заключается в том, чтобы перенести «сдвиг влево» на уровень компонентного и интеграционного тестирования и выполнять системное тестирование, только если это необходимо. Тестовые данные для тестовых сред сложных систем с сервисом самообслуживания позволяют избежать образования узких мест.
 4. Архитектура ПО: переход от монолитных архитектур к слабо связанным может сократить время выполнения (LT). Тестировщики могут помогать разработчикам, практикующим предметно-ориентированное проектирование, делясь предметными знаниями (см. ISTQB®, 2019). Тестировщикам, возможно, придется изучать такие новые технологии, как микросервисная архитектура.

В бережливом производстве существует ряд подходов (Lean Six Sigma или другие своды знаний, посвященные бережливому производству), которые также можно использовать при разработке программного обеспечения. Основная методика анализа первопричин в бережливом производстве — «пять почему». Этот подход заключается в том, чтобы последовательно задавать вопрос «Почему?» при возникновении проблемы, чтобы выйти за рамки очевидных признаков и выявить первопричину. Две другие часто используемые техники — диаграммы Парето и «рыбья кость».

Иногда статические модели (например, диаграмма «рыбья кость») дают недостаточно углубленные данные, чтобы понять первопричины проблем в динамических системах. Известный технический пример — дефекты, возникающие нестабильно, которые очень трудно обнаружить и отследить их причины.

Диаграмма причинно-следственных связей— инструмент визуализации структуры обратной связи систем. Эту диаграмму можно использовать для моделирования технических систем, а также систем, построенных на основе взаимодействия с пользователем. Диаграмма причинно-следственных связей берет свое начало от блок-схем, которые использовались для анализа производственной динамики.

3.2.3 Диаграмма причинно-следственных связей

Как упоминалось в начале п. 3.2, системное мышление имеет решающее значение для понимания, анализа и модификации сложных систем, таких как поток создания ценности, отдел в организации или системный ландшафт. Один из способов это сделать —использовать диаграмму причинно-следственных связей, которую иногда называют «системной моделью» (Larman and Vodde, 2016).

Диаграмма причинно-следственных связей — инструмент мышления, позволяющий визуализировать и анализировать причинно-следственные связи и циклы обратной связи в системе. Он показывает, как различные переменные влияют друг на друга и создают усиливающие или балансирующие циклы.

Преимущество диаграммы причинно-следственных связей заключается в возможности с ее помощью выявлять неочевидные причины и последствия, а также их взаимосвязь в более масштабных системах. Это позволяет выйти за пределы непосредственных, видимых симптомов и тем самым найти более эффективные и долгосрочные решения проблем. В отличие от других,

простых методов анализа первопричин, диаграмма причинно-следственных связей может включать в себя детали, которые помогают объяснить сложность системы: например, задержки в обратной связи и то, как цели, которые преследует организация, влияют на систему. В последнем случае наиболее эффективным и действенным решением может стать пересмотр цели. Другие преимущества диаграммы причинно-следственных связей:

- Легко фиксировать работу, так как требуются только средства рисования
- Простота и точность
- Прослеживаемость динамики систем
- Формирование совместного представления о сложных проблемах
- Выявление и отражение моделей мышления отдельных лиц или групп
- Сообщение информации о важных причинно-следственных связях, которые могут вызывать проблемы

Диаграмма причинно-следственных связей состоит из четырех основных элементов:

- переменные;
- связи между переменными;
- знак плюс или минус на связях, которые показывают, как взаимосвязаны переменные;
- маркеры цикла, которые показывают, какой тип поведения система будет воспроизводить (Sterman, 2000).

В диаграмме причинно-следственных связей используются различные условные обозначения. На рисунке 3.1 приведен их пример. На рисунке 3.2 показан типовой пример диаграммы причинно-следственных связей для отображения условных обозначений. Конкретные примеры приведены ниже.

Для создания диаграммы причинно-следственных связей важно иметь команду сотрудников с разными взглядами на проблему или систему. Основные этапы работы, которые следует проходить повторно по мере развития дискуссии:

1. Определение переменных
2. Определение причинно-следственных связей между переменными
3. Описание влияния переменных друг на друга
4. Дополнение диаграммы другими факторами, влияющими на систему (например, задержки и цели)
5. Выявление и описание усиливающих и балансирующих причинно-следственных связей
6. Определение возможных мер по решению проблемы

	<p>Переменная. Важный аспект системы. Как правило, что-то, что поддается количественной оценке: например, скорость (скорость поставки) функциональных компонентов, качество кода, количество дефектов.</p>
	<p>Причинная связь. Показывает, что существует связь между переменными: например, если количество дефектов увеличивается, то объем потерь тоже увеличивается, и наоборот.</p>
	<p>Знак плюс (+). Показывает, что изменение одной переменной приводит к изменению второй переменной в том же направлении: например, если количество тестировщиков, доступных для работы, уменьшится, производительность организации тоже уменьшится.</p>
	<p>Знак минус (-). Показывает, что изменение в первой переменной вызывает изменение в противоположном направлении во второй переменной: например, если количество опытных тестировщиков уменьшается, то количество обнаруживаемых дефектов увеличивается, и наоборот.</p>
	<p>Задержка. Если между изменением переменной и влиянием на зависимую переменную имеется значительная задержка времени, то на стрелках добавляется обозначение «ЗАДЕРЖКА».</p>

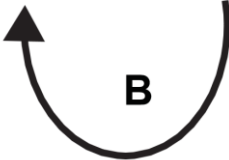
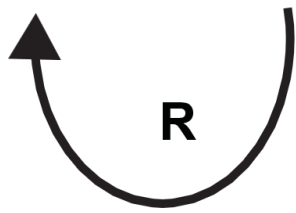
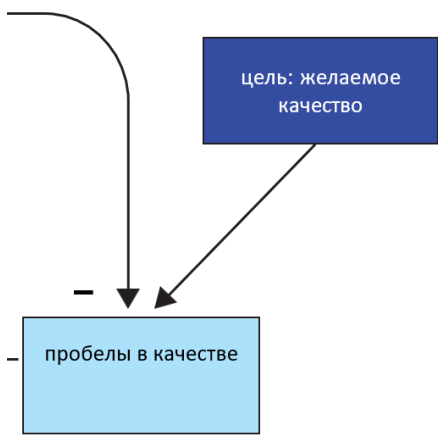
	<p>Баланс (B). Причинно-следственные связи в цикле поддерживают баланс. Циклы с нечетным числом знаков минус являются балансирующими.</p>
	<p>Усиление (R). Причинно-следственные связи внутри цикла провоцируют экспоненциальный рост. Циклы с четным числом знаков минус являются усиливающими.</p>
	<p>Цель. Результат, которого нужно достичь. Цели есть у команд, сотрудников, сложных систем и организаций</p>

Рисунок 3.1 Обозначения диаграмм причинно-следственных связей

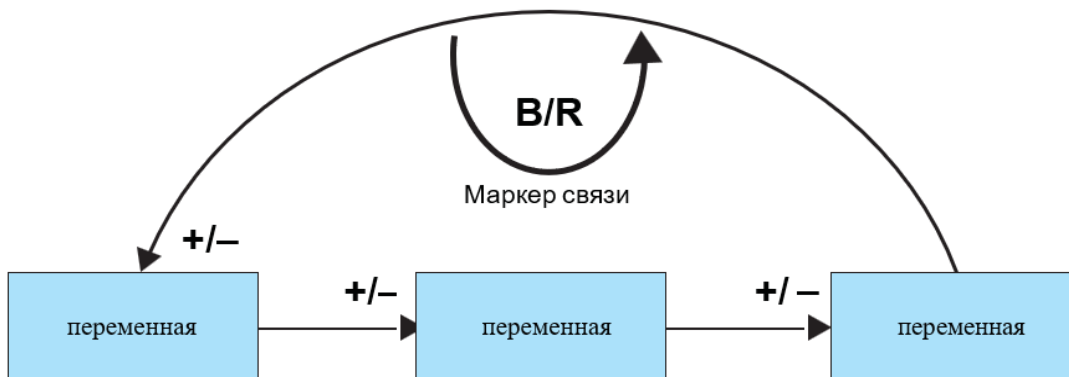


Рисунок 3.2. Типовой пример использования условных обозначений для диаграммы причинно-следственных связей

Для визуализации сложной системы может потребоваться несколько итераций. Один из важных аспектов — разделение эффектов на категории данных реальной жизни. В противном случае модель только подкрепит существующие модели мышления, но не покажет их недостатки.

Рекомендации, которые помогают визуализировать диаграмму причинно-следственных связей в сжатой и содержательной форме:

- Выберите содержательные имена переменных (используйте имена существительные, переменные, представляющие количества, которые могут изменяться во времени; выберите понятные варианты для названий переменных)
- Включите помимо ожидаемых результатов возможные, непреднамеренные последствия
- Включите цели (например, короткий цикл, в описании которого указано, что «меры по совершенствованию качества» совершенствуют «качество», а «качество» уменьшает «количество мер по совершенствованию качества», может вызывать непонимание). Если добавить «пробелы в качестве» и «желаемое качество» в диаграмму на **рисунке 3.3**, можно подчеркнуть, что «качество» уменьшает «пробелы в качестве», а «пробелы в качестве» является триггером «мер по совершенствованию качества». На **рисунке 3.4** поле «желаемое качество» добавляется вне цикла, чтобы показать, что его не следует изменять в цикле качества.

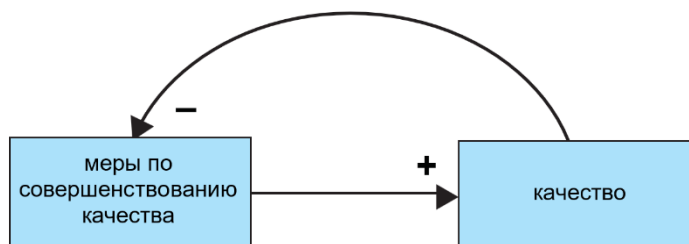


Рисунок 3.3 Пример простой диаграммы причинно-следственных связей без заданной цели

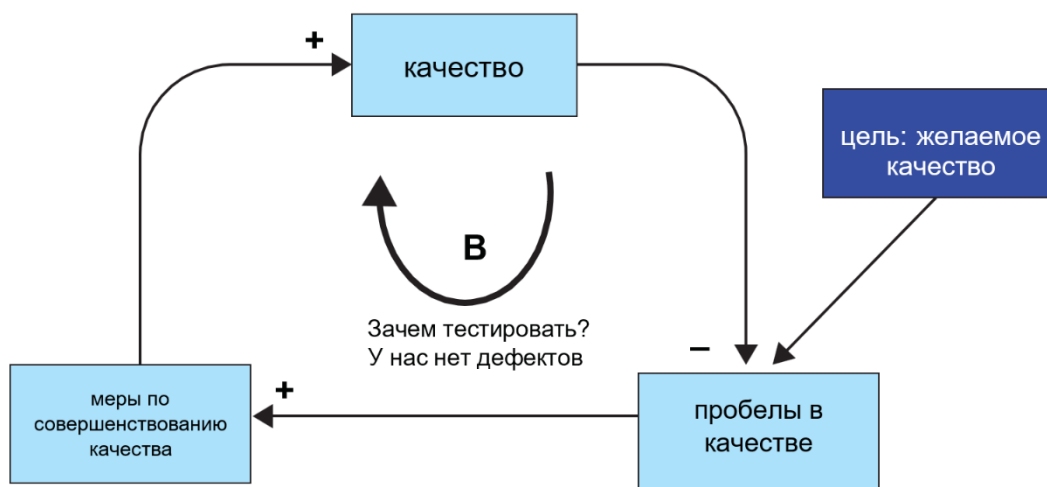


Рисунок 3.4 Пример диаграммы причинно-следственных связей с заданной целью

- Может быть полезно различать воспринимаемые и фактические состояния («воспринимаемое качество» и «фактическое качество»).
- Может быть полезно начать с переменных, суммирующих несколько аспектов, для первого взгляда (например, «зрелость автотестирования» может быть начальной переменной, а затем ее можно разделить на «степень автотестирования», «зрелость среды тестирования» и «количество доведенных до автоматизма навыков автоматизации тестирования»).
- При необходимости добавьте дополнительные, более крупные подциклы для оценки долгосрочных или краткосрочных последствий (например, «независимая оценка тестирования» совершенствует «локальное качество», но, как показано на **рисунке 3.5**, дополнительный путь «воспринимаемое давление в командах» повышает «сокрытие проблем», что снижает «общее качество», поскольку первопричины скрываются и поэтому становятся менее заметными для организации).

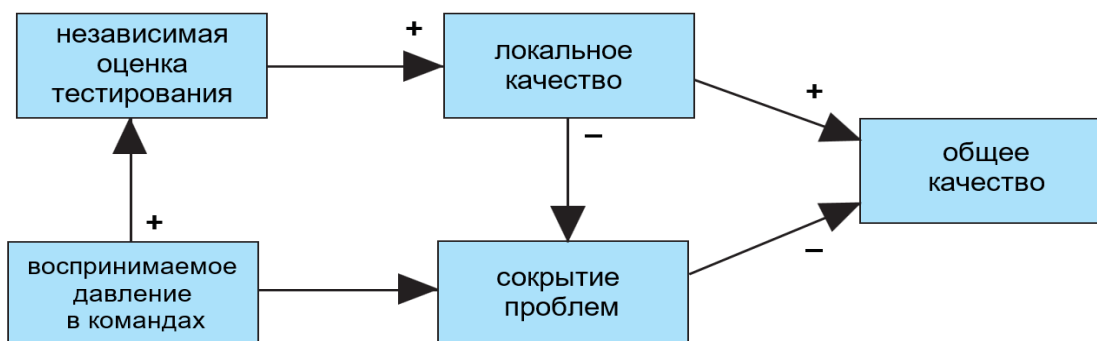


Рисунок 3.5 Пример диаграммы причинно-следственных связей с краткосрочными и долгосрочными последствиями

- Если ссылка требует слишком много объяснений, ее можно уточнить, добавив дополнительные переменные (например, если неясно, почему «рыночный спрос» снижает «качество», можно добавить новую переменную «требование скорого релиза»).
- Диаграмма причинно-следственных связей должна в первую очередь отражать подлинные причинно-следственные связи (например, следует избегать констатации, что «количество сессий тестирования» повышает «продажи продукции». С другой стороны, можно утверждать, что «количество возможностей для выявления и устранения пробелов в тестировании», обнаруженных в процессе разработки, увеличивает «общее количество сценариев тестирования», с одной стороны, и повышает «качество продукции» и, следовательно, «продажи», с другой).

При сборе и анализе результатов общей или командной ретроспективы легко попасть в ловушку локальной оптимизации. Например, если для сбора и анализа результатов используется подход «снизу вверх», порой можно забыть о том, что система не является простой совокупностью отдельных частей. Основное внимание следует уделять системности (организация, ориентированная на получение ценности, масштабная система).

Диаграмму причинно-следственных связей можно использовать для гибких ретроспектив разного типа (например, как с участием нескольких команд, так и общих ретроспектив), потому что каждая ретроспектива должна быть сосредоточена на системе (Larman and Vodde, 2016). Диаграмма причинно-следственных связей концептуально проста, но ее нелегко применять без соответствующего опыта и сопровождения.

4 Ссылки на материалы

[Arnold and Wade, 2007] Arnold, Ross, D. and Jon P. Wade. “A definition of systems thinking: A systems approach.” *Procedia Computer Science* 44 (2015): 669–678. Accessed on 6 May 2021. <https://doi.org/10.1016/j.procs.2015.03.050>.

[Bin Ali et al., 2015] Bin Ali, Nauman, Kai Petersen and Kurt Schneider. “FLOW-assisted value stream mapping in the early phases of large-scale software development.” *DiVa*, 12 August 2015. Accessed on 1 October 2020. <http://www.diva-portal.org>.

[Brito et al., 2020] Brito, Marlene Ferreira, Ana Luísa Ferreira Andrade Ramos and Paula Carneiro. “The eighth waste: Non-utilized talent.” *Research Gate*, 28 April 2020. Accessed on 5 April 2021. https://www.researchgate.net/publication/340978747_THE_EIGHTH_WASTE_NON-UTILIZED_TALENT.

[Gartner, 2018] Gartner. “DevOps and cloud speed are driving the end of QA as we know it.” Stamford, 13 August 2018.

[Goldratt and Cox, 2004] Goldratt, Eliyahu M. and Jeff Cox. *The Goal*. 3rd Edition. Oxford: Gower Publishing Ltd, 2004.

[ISTQB®, 2011] ISTQB®. “Improving the Testing Process.” International Software Testing Qualifications Board, 2011. Accessed on 18 October 2021. <https://www.istqb.org/component/jdownloads/send/12-expert-level-documents/75-expert-level-syllabus-improving-the-testing-process-2011.html>.

[ISTQB®, 2018] ISTQB®. “Certified Tester Foundation Level Syllabus.” International Software Testing Qualifications Board, 2018. Accessed on 6 May 2021. <https://www.istqb.org/certification-path-root/foundation-level-2018.html>.

[ISTQB®, 2019] ISTQB®. “Certified Tester, Advanced Level, Agile Technical Tester.” International Software Testing Qualifications Board, 2019. Accessed on 27 October 2021. <https://www.istqb.org/downloads/category/65-advanced-level-agile-technical-tester.html>.

[Kotter, 2012] Kotter, John P. *Leading Change*. Boston: Harvard Business Review Press, 2012.

[Larman and Vodde, 2016] Larman, C. and B. Vodde. *Large-Scale Scrum: More with LeSS*. Boston: Addison-Wesley, 2016.

[Lean Enterprise Institute, 2014] Lean Enterprise Institute. *Lean Lexicon: A Graphical Glossary for Lean Thinkers*. 5th Edition. Cambridge: Lean Enterprise Institute, Inc., 2014.

[Liker and Meier, 2005] Liker, Jeffrey and David Meier. *The Toyota Way Fieldbook*. New York: McGraw-Hill, 2005.

[Prosci Inc., н.д.] Prosci Inc. *The Prosci ADKAR Model*. Prosci, n.d. Accessed on 6 May 2021. <https://www.prosci.com/methodology/adkar>.

[Scaled Agile Inc., 2019] Scaled Agile Inc. (SAFe). *Principle #2 – Apply systems thinking*, 2019. Accessed on 18 October 2021. <https://www.scaledagileframework.com/apply-systems-thinking/>.

[Scrum.org, 2021] Scrum.org. *Nexus™ Guide: The Definitive Guide to Scaling Scrum with Nexus*. January 2021. Accessed on 6 May 2021. <https://www.scrum.org/resources/online-nexus-guide>.

[Senge, 1990] Senge, Peter M. *The Fifth Discipline: The Art and Practice of The Learning Organization*. 1st Edition. New York: Doubleday Dell, 1990.

[Stave and Hopper (2007)] Stave, Krystyna and Megan Hopper. "What constitutes systems thinking? A proposed taxonomy." *ResearchGate*, January 2007. Accessed on 6 May 2021. https://www.researchgate.net/publication/255592974_What_Constitutes_Systems_Thinking_A_Proposed_Taxonomy.

[Sterman, 2000] Sterman, J. D. *Business Dynamics: Systems Thinking and Modeling for a Complex World*. New York: Mcgraw-Hill Education, 2000.

[The LeSS Company B.V., без даты] The LeSS Company B.V. *Systems Thinking*. LeSS, n.d. Accessed on 18 October 2021. <https://less.works/less/principles/systems-thinking>.

[Wikipedia, 2019] Wikipedia. *Genchi Genbutsu*. Wikimedia Foundation, Inc. 22 August 2019. Accessed on 5 April 2021. https://en.wikipedia.org/wiki/Genchi_Genbutsu.

5 Дополнительное чтение

Lenny, Jason. *Value Stream Mapping the CM Pipeline*. Lean Builds, 17 May 2009. Accessed on 6 May 2021. <https://leanbuilds.wordpress.com/2009/05/17/value-stream-mapping-the-cm-pipeline/>.

Scaled Agile, Inc. *Shared Services*. SAFe, 18 December 2019. Accessed on 6 May 2021. <https://www.scaledagileframework.com/shared-services/>.

Smalley, A. and D. K. Sobek. *Understanding A3 Thinking*. London: Taylor & Francis, 2008.

Stelter, Reinhard. "Third generation coaching: Reconstructing dialogues through collaborative practice and a focus on values." *International Coaching Psychology Review* 9 (2014): 51–66.