

МЕТОДЫ И ИНСТРУМЕНТЫ УПРАВЛЕНИЯ ПРОЕКТОМ

Обеспечение устойчивости системы управления IT-проектами

Stability Assurance in IT-Projects Management Systems

DOI: 10.12737/18007

Получено: 10 декабря 2015 г. / Одобрено: 20 января 2016 г. / Опубликовано: 17 марта 2016 г.

Зуйков К.А.

Аспирант,
Национальный исследовательский университет
«Высшая школа экономики»,
Россия, 105187, г. Москва, ул. Кирпичная, д. 33,
e-mail: kirill.zuykov@gmail.com

Zuykov K.A.

Postgraduate Student,
National Research University Higher School of Economics,
33, Kirpichnaya St., Moscow, 105187, Russia,
e-mail: kirill.zuykov@gmail.com

Аннотация

Статья посвящена анализу динамики IT-проектов с целью разработки процессов управления, обеспечивающих устойчивость этих проектов к изменению содержания. В статье жизненный цикл разработки программного обеспечения, являющейся неотъемлемой частью IT-проектов, рассматривается совместно с жизненным циклом организационных изменений, для осуществления которых предпринимаются IT-проекты. Целостное рассмотрение организационной структуры IT-проектов позволяет выявить особенности динамики таких проектов, приводящих к неустойчивости и, в частности, явления «расползания требований» как основного фактора потери устойчивости проекта и, как следствие, недостижения его целей. Для обеспечения требуемой целостности модели IT-проекты рассматриваются в контексте портфеля проектов организации как составная часть программы трансформации бизнеса. При этом программа рассматривается не в полном объеме, а в той ее части, которая имеет отношение к реализации IT-проекта и получения желаемых результатов от использования его продукта. Для обеспечения возможности мониторинга достижения целей проекта на основании метрик организационного изменения работе предложена модификация метода освоенного объема для применения в управлении IT-проектами. Проведено моделирование на основе системной динамики, иллюстрирующее механизмы возникновения проблемы «расползания требований» и ее решение с использованием разработанных методов и, таким образом, обеспечение устойчивости.

Ключевые слова: управление изменениями, IT-проекты, устойчивость проекта, расползание требований, системная динамика.

Abstract

The paper is devoted to the IT-projects dynamics analysis in order to develop management processes that assure stability of the projects to scope changes. Software development lifecycle is considered combined with one of organizational changes which are carried out by IT-projects execution. This integral study of IT-projects organizational structure allows to discover projects particularities that lead to destabilization and especially "scope creep" as a main factor of loss of stability and, as a result, failure of objectives realization. To provide required level the model integrity IT-projects are to be considered in context of organization's project portfolio as a part of business transformation program. The program in turn is studied not in its entire scope but particularly in its part relative to IT-project implementation together with obtaining desired results of its product utilization. To secure metrics-based project objectives achievement control an EVM modification for IT-projects management is presented in the article. Simulation that is also performed in the study illustrates internal causes of scope creep problem and its solution with methods developed in this study and thus project stability assurance.

Keywords: change management, IT-projects, project stability, scope creep, system dynamics.

Введение

Каждая современная компания в той или иной степени зависима от информационных технологий и поэтому вынуждена выполнять IT-проекты. Говоря об IT-проектах, мы будем рассматривать проекты, содержание которых включает в себя неотъемлемую часть разработки программного обеспечения или адаптации «коробочного» программного обеспечения в соответствии с потребностями конкретного заказчика. Данная статья посвящена вопросу устойчивости систем управления такими проектами. При этом IT-проекты рассматриваются

в контексте организационных изменений, для осуществления которых предпринимаются. Существует отличие между проектами, выполняемыми в компаниях, специализирующихся на разработке программного обеспечения, и IT-проектами, которые выполняются в организациях, для которых разработка программного обеспечения не является ключевой компетенцией. Именно последним посвящена данная статья.

IT-проекты всегда связаны с некоторым организационным изменением и выполняются для его осуществления. При этом этапы жизненного цикла проекта разделяются на две части: выполняемые на

стороне заказчика и на стороне подрядчика. Зачастую менеджмент фокусируется непосредственно на разработке программного обеспечения, упуская из виду организационную часть проекта. Таким образом, потенциальные возможности, заложенные в программный продукт, не реализуются, и к продукту выдвигаются все новые требования, реализация которых не повышает эффективности бизнес-процессов заказчика.

Проекты разработки программного обеспечения подвержены изменениям в процессе выполнения, которые оказывают существенное влияние на сроки, стоимость и достижение поставленных целей. В качестве основной проблемы, указываемой практически всеми специалистами, участвовавшими в своей практике в *IT*-проектах, можно назвать так называемое «расползание требований» [13]. Для успешного выполнения *IT*-проектов требуются компетенции в области разработки программного обеспечения, владение процессами и инструментами, используемыми в этих проектах. Основная проблема состоит в том, что компании, рассматриваемые в данном исследовании, этими компетенциями, как правило, не обладают, что обуславливает множество изменений в содержании проекта по мере того, как заказчик погружается в тему разработки. Расползание требований оказывает существенное влияние не только на сроки и стоимость проекта, но и на его качество, так как задействуются такие механизмы, как снижение мотивации при длительной разработке, а также увеличение числа дефектов, вносимых программистами, работающими под постоянным давлением сжатых сроков [26]. При этом исследования показывают, что увеличение трудозатрат на планирование не приводит к уменьшению длительности проекта [6].

При этом Эйзенхардт и Табрици указывают на то, что полученный результат не относится к компаниям, для которых разработка программного обеспечения является ключевой компетенцией. То есть, приложив дополнительные усилия к планированию проекта, компании, специализирующиеся на разработке ПО, могут снизить трудозатраты на выполнение проекта и повысить качество. В то же время компаниям, которые заказывают разработку ПО для того, чтобы автоматизировать свои бизнес-процессы, создать дополнительную ценность своих традиционных продуктов или реализовать новую бизнес-модель, снизить влияние неопределенности за счет более тщательного планирования, оказыва-

ется, как правило, невозможно, и трудозатраты на планирование только увеличивают длительность и стоимость проекта.

Традиционные подходы к управлению *IT*-проектами не приводят к решению проблемы расползания требований и, как следствие, роста длительности и затрат проекта. Более того, *IT*-проекты, заказчиком которых являются организации, не специализирующиеся в разработке программного обеспечения, имеют высокие риски недостижения поставленных целей. В этих условиях представляется актуальной разработка системы управления *IT*-проектами, устойчивой к изменениям содержания. Именно это и является фокусом данной работы: в ней разрабатывается система управления *IT*-проектом, но акцентируются только те ее параметры, процессы и инструменты, которые позволяют достигать поставленных целей в условиях неопределенности требований и, как следствие, вероятных изменений в содержании.

Проблема устойчивости в управлении проектами

Потребность в управлении изменениями возникает практически во всех проектах. Целью мероприятий по управлению изменениями является удержание последствий изменений в проекте в области допустимых для заказчика потерь. Так, Ципес и Товб [35] рассматривают потери проекта в трех измерениях: по времени, ресурсам и поставляемым продуктам.

Брем и Маркус организационно разделяют *IT*-проект на часть, выполняющуюся на стороне заказчика, и часть, выполняющуюся на стороне подрядчика [4]. Жизненный цикл подпроекта, выполняемого на стороне подрядчика [21], включает в себя, как правило, следующие этапы:

- анализ;
- архитектура;
- дизайн;
- конструирование;
- интеграция;
- тестирование.

Данные этапы являются перекрывающимися, в зависимости от подхода к поставке продукта проекта могут быть итерационными, инкрементными, а также предполагать различный хронологический порядок этапов жизненного цикла.

Подпроект, выполняющийся на стороне заказчика, включает следующие этапы жизненного цикла, подробно описанные в литературе, посвященной автоматизации бизнес-процессов [22]:

- концептуализация (бизнес-кейс);
- конфигурация (подробное описание состава продукта);
- развертывание;
- передача в операционную деятельность.

Обобщенный жизненный цикл *IT*-проекта с выделением этапов, выполняемых на стороне заказчика и подрядчика, представлен на рис. 1, где последовательность этапов является условной. Как будет продемонстрировано в работе далее, зачастую интеграцию и тестирование эффективно начинать непосредственно после разработки архитектуры и выполнять непрерывно до приемки разработанного продукта. Расположение этапов жизненного цикла в последовательности на представленном рисунке в чем-то аналогично связи «окончание—окончание», применяемой при построении сетевых диаграмм, т.е. последующий этап не может закончиться ранее, чем завершится предыдущий. Этап развертывания в различных проектах может выполняться командой подрядчика или командой заказчика. Более того, программный продукт может развертываться и на вычислительных мощностях, предоставляемых заказчику подрядчиком. Это не оказывает существенного влияния на управление изменениями в проекте, поэтому в данной работе на этом внимание не акцентируется.

Поскольку, как говорилось выше, *IT*-проекты всегда связаны с организационными изменениями, они подвержены действию неопределенности, присущей таким проектам. Как отмечают Ципес и Кузьмищев, «цели проектов организационных изменений, как правило, не являются четко определенными или могут изменяться. С другой стороны, методы реализации проекта часто либо изначально не определены, либо нуждаются в постоянном уточнении по ходу выполнения работ» [34]. Кроме того, этапы

жизненного цикла *IT*-проекта, выполняемые на стороне заказчика и подрядчика, существенно отличаются по терминологии, применяемому инструментарию и требуют различных компетенций от команды проекта. Данная проблема усложняет коммуникацию между командами проекта на стороне заказчика и подрядчика и не позволяет определить содержание проекта на начальных стадиях таким образом, чтобы минимизировать изменения в процессе разработки.

В этих условиях менеджмент должен направлять усилия, очевидно, не столько на разработку более качественного плана проекта, сколько на реализацию системы управления проектом, устойчивой к изменениям содержания. Под устойчивостью будем понимать свойство системы управления проектом при возникновении расхождения с планом создавать управляющие воздействия, обеспечивающие завершение проекта с отклонением от поставленных целей на приемлемом для заказчика уровне. Для построения устойчивой системы управления проектом потребуется изучить драйверы изменений содержания и определить движущие силы проекта и факторы, тормозящие его реализацию. Их выявление является необходимым условием перехода от интуитивного принятия решений о корректирующих действиях в проекте к принятию решений, основанному на фактах [31].

При этом не следует отрицать важность планирования как такового, так как процессы планирования будут являться существенной составляющей устойчивости системы управления. Современные исследования о методах планирования предлагают подходы, позволяющие построить оптимальное расписание проекта в условиях неопределенности [10; 28], а также ищут возможности для построения расписаний, отличающихся свойством надежности

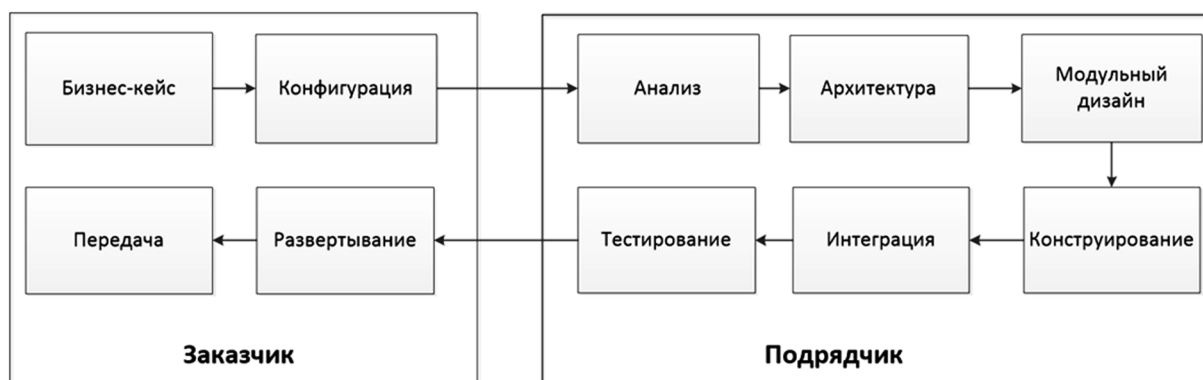


Рис. 1. Жизненный цикл проекта разработки программного обеспечения

и устойчивости по отношению к отклонениям в выполнении работ проекта [8; 9; 16].

Исследования устойчивости проектов, как правило, относятся к устойчивости расписания, т.е. в них рассматриваются вопросы построения такого расписания, при котором в случае возникновения задержек в выполнении работ проект вследствие наличия правильно рассчитанных буферов вернется со временем к выполнению в соответствии с первоначально определенным расписанием, а значит, завершится в срок. Рассматривают при этом следующие варианты поведения проекта:

- после окончания действия возмущающего фактора отклонение сокращается и выполнение работ возвращается к первоначальному расписанию;
- отклонение, возникшее вследствие задержки работы, не уменьшается и не увеличивается до конца проекта;
- возмущение вызывает увеличивающееся со временем отставание, приводя к изменению длительности проекта более чем на величину первоначальной задержки.

Первый вариант расписания проекта в некоторых случаях называют положительно устойчивым, второй — нейтрально устойчивым, третий — неустойчивым [25].

Подход к обеспечению устойчивости проекта, основанный на построении устойчивого расписания, может во многих случаях гарантировать соответствие длительности проекта плановой величине, однако только в том случае, если отклонение от выполнения по плану обусловлено не изменением содержания проекта. Для ИТ-проектов между тем изменение содержания является, как говорилось выше, одной из основных причин возникновения отклонений от выполнения по плану. Таким образом, построение устойчивого расписания может рассматриваться только как один из инструментов построения устойчивой системы управления проектом, но никак не единственный.

Кроме того, выполнение плана по длительности проекта является не единственным показателем качества управления проектом. Больше того, соблюдение ограничений классического проектного треугольника не гарантирует успешности организационного изменения, для осуществления которого выполняется ИТ-проект. Система управления ИТ-проектом должна, таким образом, включать процессы мониторинга влияния, оказываемого разрабатываемым продуктом на целевое организационное

изменение, и принятия решений о внесении изменений в содержание при отклонении выявленного влияния от желаемого. В подходе, разработанном в данной работе, для этого используется непрерывный мониторинг метрик продукта.

В данной работе ИТ-проект рассматривается в контексте организационного изменения, для осуществления которого предпринимается. Если бы цели проекта формулировались в терминах разработки программного продукта и не привязывались к метрикам изменяемых бизнес-процессов, то достаточным было бы рассмотрение вопроса устойчивости проекта в разрезе устойчивости расписания. И в этом случае следовало бы обратить внимание на метод критической цепи [32] и его производные. Вопрос обеспечения устойчивости при этом сводился бы к вопросу о правильном определении размера буферов и об их эффективной расстановке в расписании проекта.

Этот подход, однако, неприемлем для целей работы, поскольку не учитывает возможные изменения содержания проекта, обусловленные корректирующими действиями по результатам мониторинга достижения плановых значений метрик продукта, связанных притом с параметрами осуществляемого организационного изменения.

Для исследования устойчивости системы управления проектом необходимо проанализировать факторы, которые воздействуют на динамику проекта, и существующие в системе управления положительные (усиливающие) и отрицательные (балансирующие) обратные связи. Зачастую на проект оказывают влияние одновременно и усиливающие обратные связи, и балансирующие, и в этом случае вопрос об устойчивости системы управления проектом оказывается неочевидным. В качестве примера можно рассмотреть ситуацию в проекте организационного изменения или в программе трансформации бизнеса, когда руководство стремится осуществить быстрые преобразования для получения ранних результатов, что должно снизить сопротивление участников проекта, но при этом участники проекта могут быть не готовы к быстрым переменам, и поэтому их сопротивление будет повышаться.

Место ИТ-проектов в портфеле организации

Несмотря на то что проекты в сфере информационных технологий связаны с некоторыми организационными изменениями, современные научные публикации, равно как и публикации практической

направленности, как правило, предпочитают рассматривать вопросы управления изменениями и управления проектами разработки программного обеспечения отдельно. Проблемы ИТ-отрасли при этом рассматриваются в отрыве от проблем развития бизнес-процессов и организационных изменений [33]. Это во многом закономерно, так как разработка программного обеспечения и осуществление организационных изменений требуют у команды проекта наличия совершенно различных компетенций. Исключение составляют публикации об управлении ИТ-услугами (*IT service management, ITSM*) на основе библиотеки инфраструктуры ITIL [12] и сервисно ориентированном управлении технологиями в организациях [5]. Однако *ITSM* и сервисно ориентированное управление технологиями имеют лишь опосредованное отношение к управлению проектами.

Значительная часть исследований, посвященных информационным технологиям, рассматривает вопросы гибкого управления разработкой программного обеспечения (*Agile Development*) и управления проектами (*Agile Project Management*). Эффективность гибких методологий в управлении проектами разработки программного обеспечения во многом обусловлена тем, что именно гибкие методологии имеют инструменты, позволяющие справиться с явлением «расползания требований».

Развитием этих исследований являются исследования, посвященные вопросу организационной гибкости (*organizational agility*), т.е. построению таких организационных структур, которые позволяют организации быстро адаптироваться к постоянным изменениям среды бизнеса [19]. Относительно этих исследований заслуживает внимание тот факт, что, несмотря на то, что концепция организационной гибкости является продолжением и развитием гибкой разработки программного обеспечения и управления проектами, распространяя те же принципы, которые применяются на уровне проекта, на уровень организации, эти исследования, тем не менее, полностью теряют из вида ИТ-составляющую, фокусируясь исключительно на построении адаптивных процессов [14] и адаптивных организационных структур [2].

По отношению к ИТ-проектам организации-разработчики могут быть классифицированы на продуктовые, занимающиеся разработкой так называемых «коробочных» продуктов, т.е. продуктов, которые могут многократно тиражироваться и прода-

ваться различным потребителям в неизменном виде или с минимальными доработками под заказчика, и организации, занимающиеся заказными разработками, выполняемыми единственной раз под конкретного заказчика. Во втором случае продукт также может тиражироваться, если он является составной частью тиражируемого продукта заказчика.

Соответственно, заказчик может иметь в портфеле проекты внедрения «коробочных» продуктов или заказной разработки ИТ-продуктов для реализации определенного бизнес-кейса. Наличие ИТ-проектов в портфеле заказчика всегда предполагает осуществление некоторого организационного изменения или целого комплекса организационных изменений, составляющих программу трансформации бизнеса.

Если ИТ-проект реализуется для осуществления организационного изменения, то и его измеримые цели могут быть выражены посредством указания величины требуемого изменения. Эти цели могут быть связаны с уменьшением трудозатрат, стоимости или затрат времени на выполнение бизнес-процессов потребителя продукта ИТ-проекта либо с повышением качества оказываемых компанией услуг, что выражается метриками и показателями этих услуг.

Наибольшую сложность при постановке цели ИТ-проекта составляет переход от параметров реализуемого при поддержке ИТ-проекта организационного изменения к конкретным требованиям разрабатываемого продукта. Этот переход существенно упрощается при использовании сервисного подхода [29], поскольку метрики ИТ-сервисов могут быть непосредственно связаны с параметрами бизнес-процессов потребителя ИТ-услуги. На применении сервисного подхода основываются и разработанные в данной работе модели и методы.

Для того чтобы описать систему управления ИТ-проектом, необходимо рассмотреть:

- организационную структуру проекта;
- взаимосвязи с другими проектами и связь со стратегией организации;
- роли в проекте и ответственность;
- процессы управления проектом;
- метрики продукта и метрики процесса.

Данная работа фокусируется на управленческих аспектах, обеспечивающих устойчивость проекта к изменениям содержания. Несмотря на важность для достижения целей ИТ-проекта определения процессов разработки всех групп (инициация, планирова-

ние, исполнение, мониторинг и управление, закрытие) [21], в данной работе они будут рассматриваться только в том случае, если представляют интерес для целей исследования.

В соответствии с идеологией управления *IT*-сервисами *ITSM* [15] выделяются следующие общие цели *IT*-проектов:

- улучшение бизнес-процессов;
- сокращение затрат предприятия;
- увеличение использования информации и аналитики в принятии решений;
- повышение продуктивности трудовых ресурсов предприятия;
- привлечение новых и сохранение существующих потребителей;
- создание инновационных продуктов и услуг.

Д. Кнапп, кроме того, соотносит эти цели с целями менеджмента бизнес-процессов (*BPM*):

- приведение бизнес-процессов в соответствие с потребностями клиентов;
- улучшение способности организации достигать бизнес-целей;
- повышение продуктивности и эффективности бизнес-процессов;
- обеспечение гибкости, поддержка инноваций, освоение новых технологий [15].

Организационная структура *IT*-проекта, таким образом, включает команду проекта со стороны заказчика и команду проекта со стороны подрядчика. Команда проекта со стороны заказчика реализует содержание организационного изменения, поддержанного *IT*-проектом. Со стороны подрядчика реализуется, в свою очередь, проект разработки программного обеспечения (или доработки универсального решения для удовлетворения потребностей конкретного заказчика). Для постановки целей *IT*-проекта необходимо понимать, каким образом определяются цели самого организационного изменения. Они, в свою очередь, следуют из бизнес-кейса, который, в свою очередь, может быть разработан (но не обязательно) в рамках программы трансформации бизнеса [27].

Если организационное изменение осуществляется в рамках программы, то продукт проекта должен поддерживать реализацию целей программы. При этом сама программа предпринимается для того, чтобы преодолеть некоторый качественный разрыв в показателях бизнеса по сравнению с требуемым для занятия компанией определенного положения на рынке, например, стать лидером или, будучи

последователем, достичь определенных показателей эффективности бизнеса. На рис. 2 представлена схема создания бизнес-ценности посредством выполнения проектов [23]. Эта схема хорошо иллюстрирует механизм осуществления организационных изменений посредством выполнения *IT*-проектов. Продукт проекта разработки программного обеспечения (собственно программный продукт) передается команде проекта на стороне заказчика, управляющей организационными изменениями, благодаря которым реализуются запланированные выгоды, и как следствие создается некоторая бизнес-ценность. В данной работе, однако, *IT*-проект рассматривается совместно с организационными изменениями, так как предлагаемые методы контроля проекта основаны на метриках их результатов.

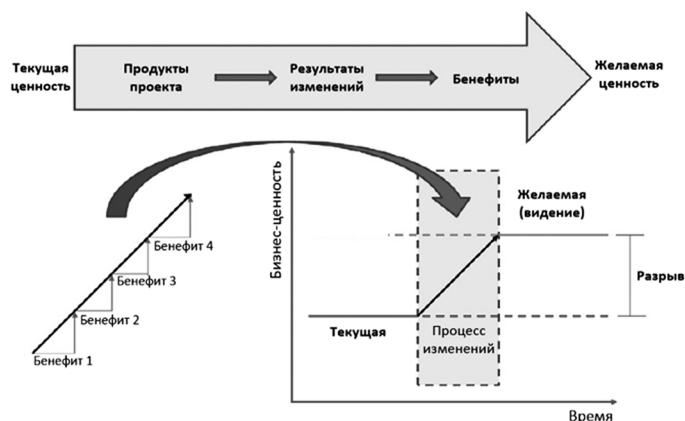


Рис. 2. Схема создания бизнес-ценности [23]

В том же источнике приводится блок-схема реализации выгод (бенефитов) программы (рис. 3). Данная работа посвящена управлению изменениями в *IT*-проектах, фокус исследования находится поэтому на проектном управлении, но проектное управление рассматривается совместно с организационным изменением. Блоки, исследуемые в данной статье, таким образом, выделены на рисунке цветом. Другие блоки тоже, однако, имеют важность, так как именно они обуславливают наличие изменений содержания, которые в данной работе исследуются с целью построения устойчивой системы управления *IT*-проектом. Хотя настоящая работа не ставит целью разработку модели, описывающей трансформационную программу в полном объеме, ограничиваясь лишь рассмотрением входящих в программу проектов организационных изменений, связанных с разработкой программного обеспечения, выводы, полученные в данной работе, могут быть в дальней-

шем использованы при построении моделей трансформационных программ.

Если возможно, планирование поставки продукта проекта следует вести «сверху вниз», в направлении, обратном направлению потока создания бенефитов на схеме. Тогда целевые показатели для отдельных организационных изменений и измеримые цели проектов разработки программного обеспечения будут являться следствием плана реализации выгод (бенефитов) программы. Следует, однако, отметить, что бенефиты не всегда являются измеримыми сами по себе [27], что несколько осложняет постановку целей проектов и вынуждает вводить искусственные метрики для их последующей экспертной оценки заинтересованными сторонами.

В альтернативном случае организация может выполнять проект по повышению эффективности бизнес-процессов для того, чтобы иметь возможность предоставлять потребителю услугу, соответствующую текущим требованиям рынка, или для того, чтобы получить за счет этого конкурентное преимущество. В этом случае измеримой целью IT-проекта явля-

ется достижение одного или нескольких целевых значений соответствующих показателей бизнес-процесса (его длительности, стоимости, а также показателей качества).

В организационной структуре проекта выделяется, таким образом, роль ответственного за реализацию бизнес-кейса. Эта роль во многом аналогична роли владельца продукта (*product owner*), являющейся ключевой в гибких методологиях управления проектами. Имеются, тем не менее, существенные отличия, которые зачастую не позволяют использовать гибкое управление непосредственно. В соответствии с задачами, возлагаемыми на эту роль, и требуемыми компетенциями, будем называть ее «архитектор организационного изменения».

Среди процессов управления проектом особенно выделим группу процессов мониторинга и управления и популярный метод контроля проекта — метод освоенного объема. Несмотря на то что метод освоенного объема позволяет отслеживать прогресс выполнения проекта одновременно в двух измерениях — по срокам и затратам, он не вполне подхо-

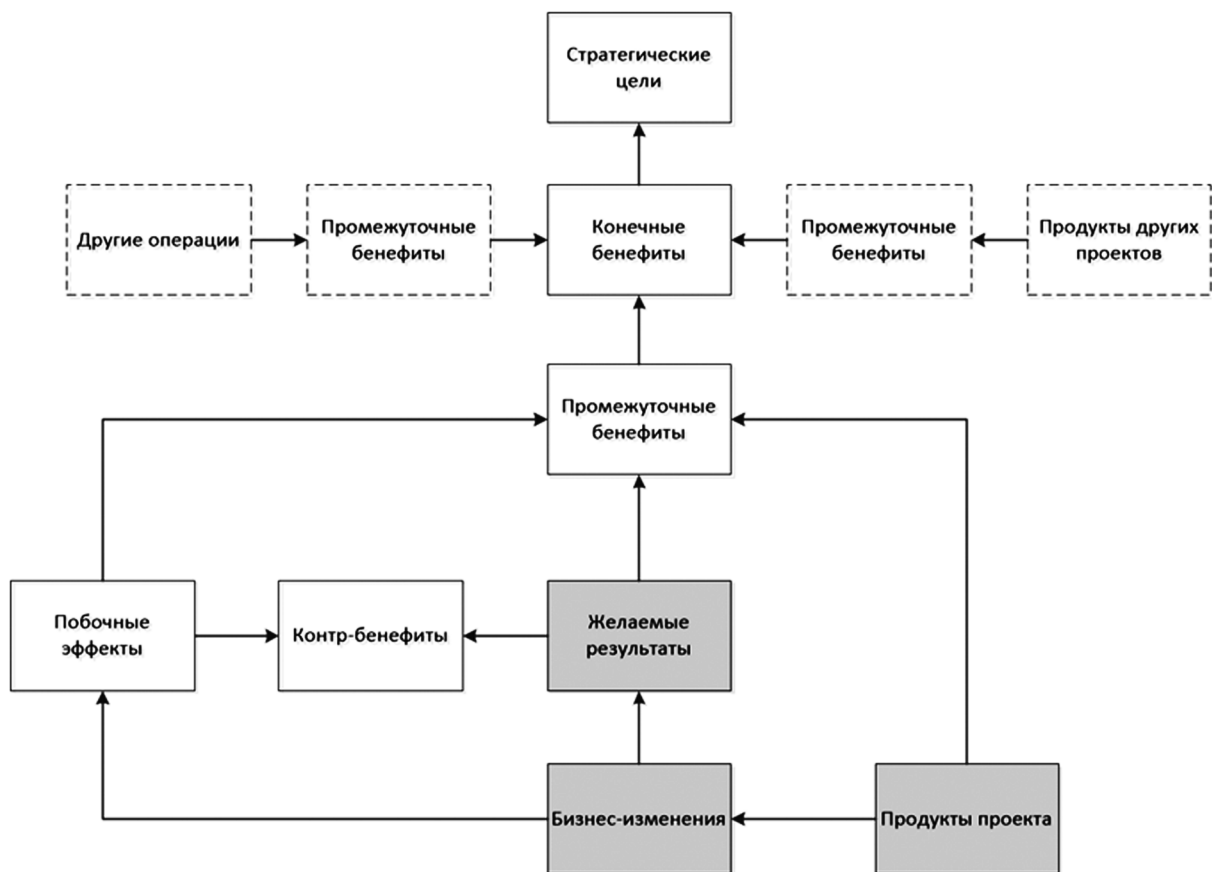


Рис. 3. Блок-схема реализации выгод программы [23]

дит для контроля *IT*-проекта как проекта организационного изменения. Дело в том, что метод освоенного объема контролирует прогресс в выполнении работ проекта и предполагает, что проект завершается, когда все работы выполнены. При этом, используя непосредственно метод освоенного объема, невозможно отследить, реализуются ли цели организационного изменения. С другой стороны, простое отслеживание метрик и показателей бизнес-процессов для мониторинга прогресса организационного изменения не дает информации о том, возникают ли отклонения в планах достижения показателей в связи с отставанием от графика или они вызваны причинами, связанными с несоответствием *IT*-продукта или его неэффективным использованием. В данной работе представлен модифицированный метод освоенного объема, лежащий в основе разработанной модели проекта и позволяющий отслеживать не только выполнение работ проекта и понесенные затраты, но и достижение планируемых показателей бизнес-процессов.

Если поставка продуктов проекта осуществляется последовательно, привязана к вехам проекта и тестирование показателей продукта можно провести при достижении той или иной вехи, при этом показатель является бинарным (или достигнут, или нет), то в использовании разработанного в работе метода освоенного объема это является излишним. В таком случае вполне достаточно применения также изложенного в настоящей работе подхода «менеджмент через тестирование». Эти условия, тем не менее, трудно достижимы по следующим причинам:

- разработка не всегда осуществляется последовательно, над проектом могут работать несколько команд, разрабатывающих параллельно части продукта. Эффекты от различных частей продукта могут складываться, при этом также возможны положительные и отрицательные синергетические эффекты;
- эффект, оказываемый разработанным программным обеспечением на бизнес-процессы, может быть отложенным, так как внедрение продукта осуществляется командой на стороне заказчика и продукт требует времени для освоения;
- показатели, по которым отслеживаются цели проекта, не всегда имеют бинарное выражение. Зачастую даже если показатель не достигает целевого значения, внесенный вклад в улучшение

бизнес-процесса может иметь ценность, и для дальнейшей оптимизации в образ продукта могут быть внесены изменения по сравнению с первоначальным планом.

Как сказано выше, для построения системы управления *IT*-проектом требуется определить метрики и показатели эффективности. Метрики и показатели, в частности, используются в предложенном в данной работе варианте метода освоенного объема. Необходимо отметить, что между метриками и показателями эффективности имеется отличие. Метрикой называется некоторое измерение рассматриваемого бизнес-процесса или продукта. Показатели эффективности же, в свою очередь, отражают выполнение или невыполнение соглашения об уровне сервиса (*SLA*) и поэтому должны иметь целевые значения, с которыми производится сравнение при оценке оказываемой услуги [30]. В *IT*-проекте могут использоваться и метрики, и показатели эффективности, но их назначение различное. Цели проекта должны быть выражены в показателях эффективности, в то время как метрики могут использоваться для отслеживания развития процессов и продуктов, а также для раннего предупреждения о наступлении рисков.

Мониторинг *IT*-проектов на основе метода освоенного объема

Организации реализуют свои стратегии, выполняя проекты (в том числе в сфере информационных технологий). При этом успех проекта не всегда гарантирует достижение стратегических целей организации. В программном управлении существует процесс поставки выгод программы, который в той или иной степени поддерживает достижение организацией стратегических целей [23]. В проектном управлении механизм, подобный процессу поставки выгод, отсутствует. Когда мы ведем речь об управлении *IT*-проектами, в силу того, если цель проекта формулируется в терминах организационного изменения, есть смысл реализовать подобный процесс в системе управления проектом. Вместе с тем в соответствии с блок-схемой на рис. 3 в данной работе не рассматривается вопрос поставки выгод, и границы проекта включают лишь получение результатов организационного изменения. Соответственно, вместо измерения реализации бенефитов будут использоваться метрики изменяемого бизнес-процесса.

Для определения метрик и целевых показателей изменяемого бизнес-процесса можно использовать технологию *GQM* (*goal — question — metric*, «цель — вопрос — метрика») [17] или иную технологию, позволяющую определить ключевые целевые показатели. Суть технологии *GQM* состоит в том, что процесс определения метрик разделяется на три этапа: а) определение целей (в нашем случае — целей организационного изменения); б) постановка вопросов ответы на которые определяют достижение целей); в) определение метрик (значения которых определяют ответы на поставленные вопросы).

Одновременно с реализацией процесса реализации изменений необходимо организовать и мониторинг, а также процесс принятия решений о внесении корректирующих действий. В данной работе предлагается использовать метод освоенного объема, дополненный возможностью мониторинга изменения метрик продукта в их привязке к выполнению работ проекта. Вообще, применению метода освоенного объема посвящено большое число публикаций, и они продолжают появляться. Это вполне объяснимо, так как метод освоенного объема чрезвычайно распространен как метод мониторинга проектов с возможностью прогнозирования сроков завершения и фактических затрат на конец проекта. Вместе с тем то, что в силу его универсальности не учитываются потребности конкретных типов проектов, открывает большие возможности для исследований. В тематике данной статьи интерес вызывает публикация, в которой анализируется применение метода освоенного объема совместно с гибким управлением проектами, а также сопоставляются показатели эффективности процесса *SCRUM* и показатели метода освоенного объема [24]. Несмотря на то что гибкое управление проектами ориентировано, в первую очередь, на быстрое создание ценности для потребителя, представленный в работе метод освоенного объема ограничивается анализом коэффициентов продуктивности по временным параметрам и стоимости (как и в классическом случае). Таким образом, он не позволяет осуществлять мониторинг процесса создания ценности для реализации корректирующих действий на уровне продукта.

Идея модифицировать метод освоенного объема таким образом, чтобы с его помощью иметь возможность также контролировать реализацию выгод, на получение которых направлен проект или програм-

ма, не нова. Так, в 2012 г. в США был зарегистрирован патент [11], который наряду с традиционными индексами продуктивности по срокам и затратам *SPI* и *CPI* предлагал ввести в метод освоенного объема индекс продуктивности по реализации выгод *BPI*:

$$BPI = \frac{AB}{PB}, \quad (1)$$

где *AB* — фактическое значение показателя реализации выгод;

PB — плановое значение показателя реализации выгод.

Действительно, такой коэффициент был бы чрезвычайно полезен и позволил бы решить проблему, обусловленную тем, что выполнение всех работ проекта или всех задач программы не гарантирует получения планируемых выгод. Вместе с тем в версии, зафиксированной в данном патенте, предложен способ определения коэффициента, использование которого ведет к другой проблеме, а именно: *BPI*. Рассчитанный таким способом коэффициент не дает возможности дифференцировать ситуации, когда выгоды не получены вследствие того, что не выполнены запланированные работы и задачи, и когда выгоды не получены в условиях выполнения проекта в соответствии с планом, что может быть вызвано тем, что продукт проекта определен неверно и его получение не ведет к реализации планируемых выгод.

Предлагается изменить подход к определению индекса продуктивности по реализации выгод. При сохранении общей формулы для определения *BPI* предлагается использовать в качестве *PB* не плановое значение переменной, выражающей выгоды на данный момент времени, а плановое значение этой переменной для текущего значения освоенного объема. Для простоты будем называть этот коэффициент также (*BPI*), но в данной работе он, во-первых, связан не с выгодами программы, а с метриками, по которым оценивается эффективность организационного изменения, а во-вторых, может распадаться на несколько отдельных показателей эффективности по числу используемых метрик (*BPI_i*).

Рассмотрим предлагаемый подход на примере. В табл. 1 приведены плановый объем и освоенный объем проекта *PV* и *EV* по периодам (в настоящее время — пятый период), а также плановое значение выгод и фактическое значение отслеживаемой метрики (*PB* и *AB*).

Таблица 1

Пример отслеживания проекта по методу освоенного объема

	0	1	2	3	4	5	6	7	8	9	10
PV	0	2	3	3,5	4,1	5	6	6,5	6,7	7,2	8,5
EV	0	0,5	0,7	1,1	1,9	2,5					
PB	0	0,3	0,5	0,7	1	1,5	2,5	3,5	5	7	10
AB	0	0,07	0,17	0,27	0,43	0,6					

На рис. 4 приведен график мониторинга проекта по методу освоенного объема. Видно, что по временным параметрам проект отстает от графика (параметр стоимости для нашей задачи не принципиален, поэтому не анализируется).

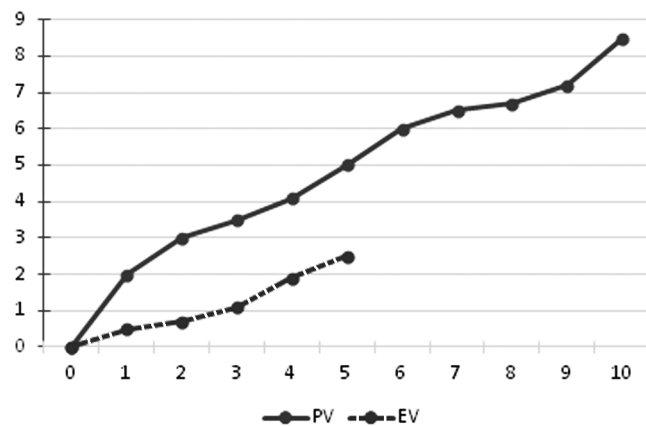


Рис. 4. Метод освоенного объема

Если рассчитать *BPI* по исходной формуле, то получим, что

$$BPI = \frac{AB}{PB} = \frac{0,6}{1,5} = 0,4, \quad (2)$$

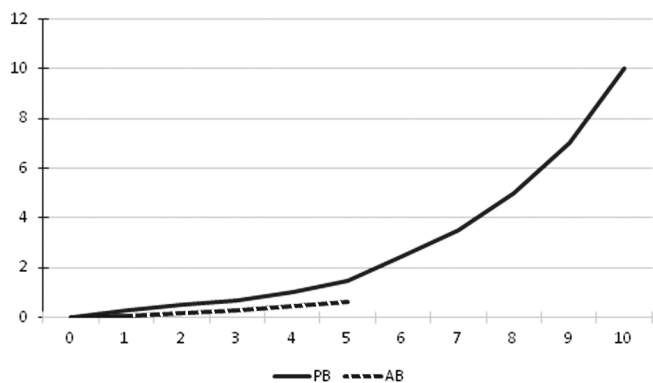


Рис. 5. Мониторинг метрики организационного изменения

Построим зависимость плановых значений метрики от планового объема.

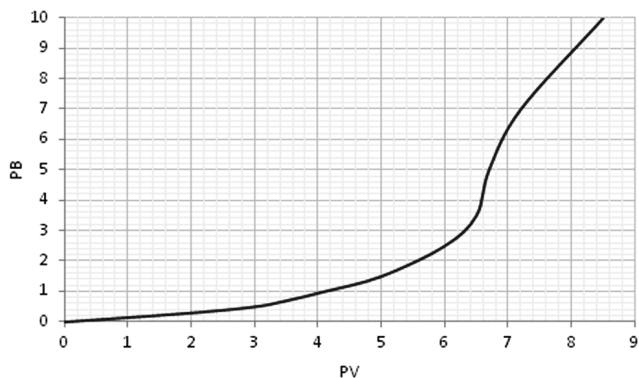


Рис. 6. Зависимость плановых значений метрики от планового объема

Нетрудно заметить, что для текущего значения освоенного объема значение переменной, отражающей реализацию выгод, должно быть равно 0,4 и, таким образом,

$$BPI = \frac{AB}{PB(EV)} = \frac{0,6}{0,4} = 1,5, \quad (3)$$

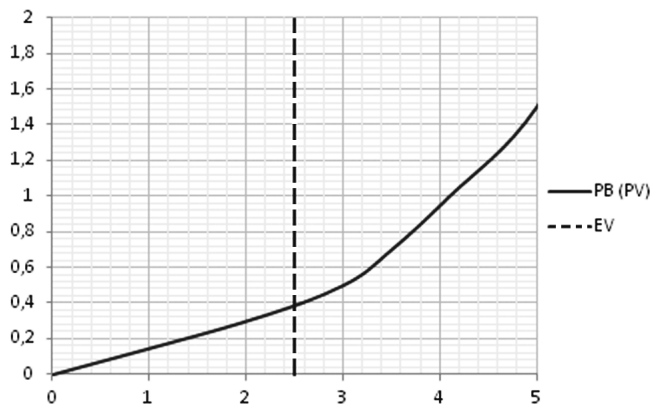


Рис. 7. Плановое значение метрики для текущего значения освоенного объема

Иными словами, при управлении данным проектом следует особое внимание уделять выполнению работ по разработке продукта, а не по реализации организационного изменения, поскольку эффективность процесса реализации организационных изменений на данном этапе выполнения проекта является удовлетворительной.

Принятие решений о корректирующих действиях при применении метода освоенного объема, таким образом, опирается на показатели эффективности управления по временным параметрам, по стоимости и по реализации выгод. При этом необходимо помнить, что отклонения от плана достижения целевых показателей могут быть обусловлены для

IT-проекта как причинами, относящимися к дизайну программного продукта (несоответствие возможностей продукта требованиям организационного изменения), так и причинами организационными, включающими в числе прочего недостаточную компетентность конкретных исполнителей, работающих с новым программным продуктом, недостатки в организации (нечеткие инструкции, отсутствие ответственных за выполнение этапов бизнес-процесса), а также, что характерно для всех проектов организационных изменений, сопротивление изменениям.

Сопротивление изменениям можно выделить как фактор, оказывающий критическое влияние на достижение целей как отдельного проекта организационных изменений, так и программы трансформации бизнеса в целом. Вместе с тем сопротивление изменениям может быть обусловлено различными факторами [20], и поэтому отсутствует какой-либо универсальный подход к управлению сопротивлением. Менеджеру проекта следует анализировать драйверы сопротивления и применять методы работы с командой и заинтересованными сторонами, соответствующие сложившейся ситуации.

На сопротивление изменениям следует обратить внимание также по следующей причине: существуют исследования, которые указывают на то, что те же факторы, которые обуславливают наличие сопротивления изменениям, обеспечивают надежность бизнес-процессов организации в периоды, когда изменения не происходят. Инвесторы и клиенты могут ценить способность организации постоянно производить продукцию определенного качества выше, чем ее эффективность. В этом также состоит существенное отличие организаций от проектных групп. «Организации обладают более высокими уровнями надежности, чем проектные группы, по двум позициям: с точки зрения одномоментного (*cross-sectional*) и межвременного (*temporal*) сравнения. Одномоментная надежность означает, что случайно выбранному из популяции организаций результату будет свойственна меньшая дисперсия, чем результату, случайно выбранному из популяций производителей другого типа. Временная надежность означает, что колебания качества (включая время получения) результата будут ниже для организаций, чем для каких-то проектных групп» [7].

Явное сопротивление изменениям является, кроме того, позитивным фактором и для самого проекта организационных изменений. Дело в том, что

поскольку будущее не может быть никому достоверно известно, людям свойственно опасаться неизвестного в будущем, и отсутствие сомнений в целесообразности изменения у большинства участников проекта является, как правило, признаком низкой вовлеченности, что впоследствии может привести к снижению эффективности организационного изменения и даже к его саботированию. Таким образом, наличие явного сопротивления изменениям позволяет менеджеру проекта инициировать диалог с заинтересованными сторонами проекта. В результате диалога можно ожидать как вовлечения заинтересованных сторон в работу над организационным изменением, так и получение обратной связи, что позволит лучше определить содержание изменения и образ IT-продукта, разрабатываемого для его осуществления.

Таким образом, можно рекомендовать следующие мероприятия при обнаружении снижения эффективности управления выгодами в проекте:

- создание рабочей группы, включающей представителей команд со стороны заказчика и подрядчика, а также пользователей разрабатываемого программного продукта, для проведения аудита бизнес-процессов и принятия решений о корректирующих действиях (организационных или связанных с изменением продукта);
- аудит бизнес-процессов для выявления фактов соответствия или несоответствия способов использования разработанного программного обеспечения вариантам использования, предусмотренным дизайном;
- в случае обнаружения несоответствия — проверка компетентности персонала, работающего с программным обеспечением, аудит обеспеченности персонала инструкциями, обучающими материалами, доступности технической поддержки;
- при необходимости — проведение повторного обучения персонала новым бизнес-процессам и работе с программным обеспечением;
- принятие решений о внесении изменений в программный продукт для упрощения и повышения эффективности работы с ним по результатам испытаний в реальных условиях;
- аудит обеспеченности бизнес-процессов требуемыми ресурсами;
- в случае обнаружения несоответствия обеспеченности бизнес-процессов ресурсами — принятие решений о добавлении ресурсов или освобождении имеющихся трудовых ресурсов от иных задач;

- сбор обратной связи от пользователей программного продукта для выявления потребностей персонала, работающего с ним, и вовлечения заинтересованных сторон в разработку продукта и внедрение новых бизнес-процессов;
- в случае возникновения утвержденного запроса на дополнение содержания определяется приоритет результата этого дополнения в общем списке, и дополнение вносится в план в соответствии с этим приоритетом. При этом менее приоритетные результаты должны быть отодвинуты на более поздний срок.

Моделирование проблемы расползания требований

Далее описывается динамическая модель IT-проекта, построенная в среде моделирования системной динамики *VenSim*. Модель основана на принципах, разработанных в статье с использованием методов системной динамики. Примечание: каждой переменной будет дано два имени. Длинное имя будет использоваться для удобства отображения в модели *VenSim*, короткое — в дифференциальных уравнениях, описывающих динамику.

Начнем построение модели с базового процесса разработки (рис. 8).

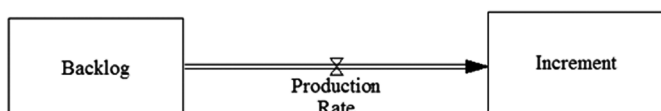


Рис. 8. Базовый процесс разработки ПО

Переменные базового процесса:

Backlog (*B*) — объем требований к продукту, который предстоит выполнить до завершения проекта, выраженный в относительных единицах трудоемкости *sp*;

Increment (*I*) — объем выполненных работ;

Production Rate (*PR*) — текущая продуктивность команды (*sp/день*).

Уравнение базового процесса разработки:

$$\frac{dI}{dt} = PR. \tag{4}$$

Данное исследование не ставит целью исследовать в полном объеме факторы, влияющие на продуктивность команды разработки, тем более что существует достаточное число публикаций, посвя-

щенных именно этому вопросу [18]. При моделировании базового процесса не будем учитывать также и неравномерность процесса разработки, обусловленную планом, т.е. будем в данной версии модели предполагать $PR = const$.

Объем требований к продукту изменяется в процессе разработки (рис. 9). На динамику генерирования новых требований влияет коэффициент неопределенности, который в данной модели привязан в соответствии с конусом неопределенности Боэма к относительному прогрессу разработки (через него выражается близость проекта к завершению).

New Features Rate (*NFR*) — темп генерирования требований к продукту (в *sp/день*). Дифференциальное уравнение изменения объема требований к продукту примет, таким образом, следующий вид:

$$\frac{dB}{dt} = NFR - PR. \tag{5}$$

Relative Progress (*RP*) — доля выполнения задач проекта по состоянию на данный момент (от 0 до 1).

$$RP = \frac{I}{I + B}. \tag{6}$$

Uncertainty Coefficient (*UC*) — коэффициент темпа генерирования проекта в связи с неопределенностью.

В соответствии с утверждением Боэма [3] для проектов разработки программного обеспечения коэффициент изменяется от 4 в начале выполнения проекта до 0 в момент завершения проекта.

$$UC = 4 \cdot (1 - RP), \tag{7}$$

$$NFR = nfr \cdot UC, \tag{8}$$

где *nfr* — некий нормирующий коэффициент, зависящий от принятой в компании или в данном проекте политики внесения изменений в содержание продукта.

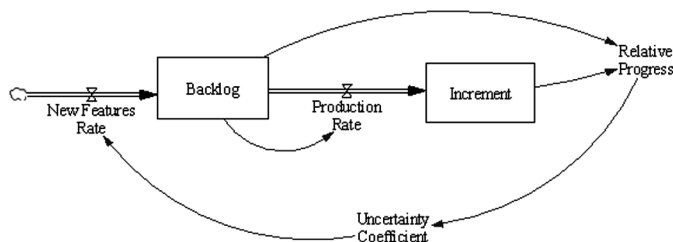


Рис. 9. Изменение объема требований

Результат моделирования (рис. 10) показывает, что в такой системе возможен существенный рост содержания проекта, но в целом система управления

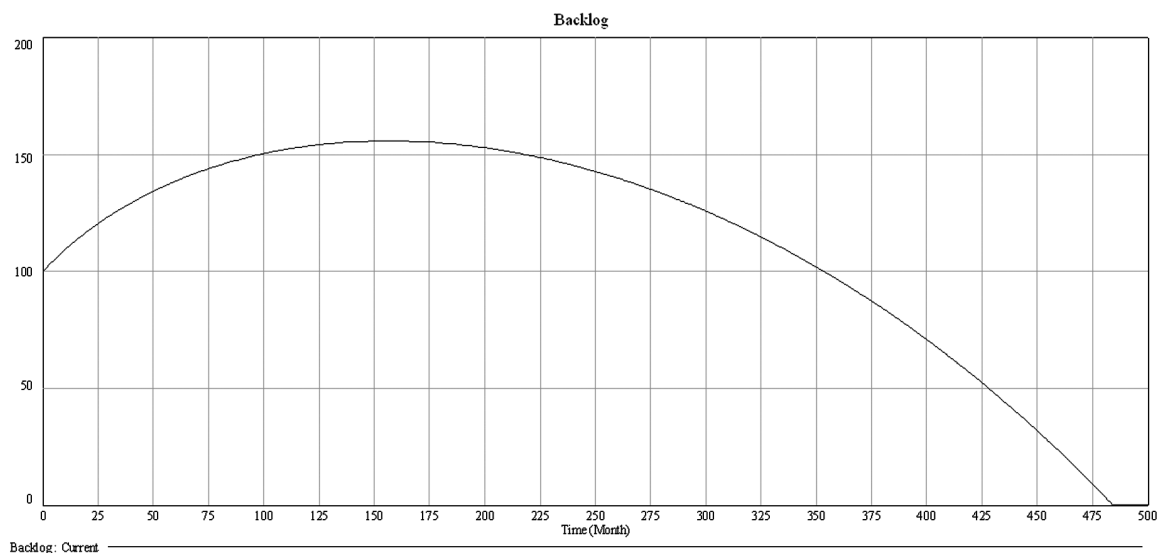


Рис. 10. Результат моделирования (объем требований)

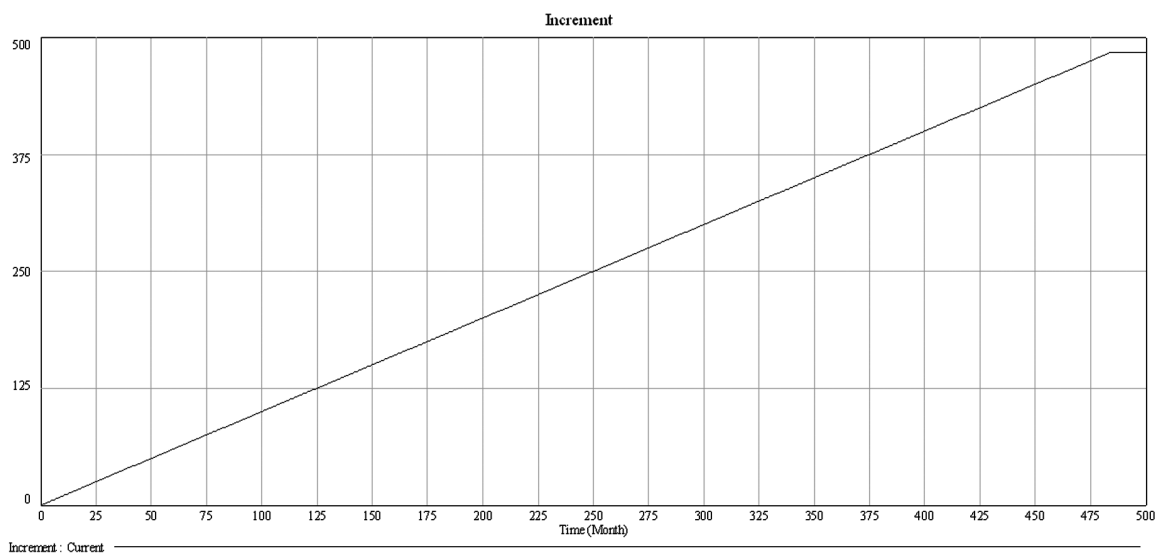


Рис. 11. Результат моделирования (инкремент)

проектом при данных параметрах демонстрирует поведение, характерное для устойчивых систем (не происходит лавинообразного создания новых требований). Поэтому проект все же завершается, хоть почти с пятикратным превышением планируемых трудозатрат (рис. 11).

В первой версии модели продуктивность команды была зафиксирована, однако она изменяется в процессе выполнения проекта разработки. Из множества факторов, влияющих на продуктивность [1; 18] выделим следующие (они более других связаны прогрессом в проекте):

- *Dev Motivation Coef (DMC)* — коэффициент продуктивности, связанный с мотивацией команды. Продуктивность команды разработчиков суще-

ственно зависит от их вовлеченности в проект, и вовлеченность при значительном превышении сроков проекта существенно снижается. В данной модели мы будем использовать экспоненциальную модель снижения продуктивности команды по времени (рис. 12);

- *Support Fraction (SF)* — доля рабочего времени, уделяемого командой разработки на поддержку ранее разработанных частей программной системы. Эта доля с некоторым коэффициентом пропорциональна прогрессу в реализации продукта. Рост трудозатрат на поддержку обусловлен тем, что с прогрессом разработки растет сложность программной системы, что обуславливает вероятность ошибок, в том числе ошибок интеграции модулей системы.

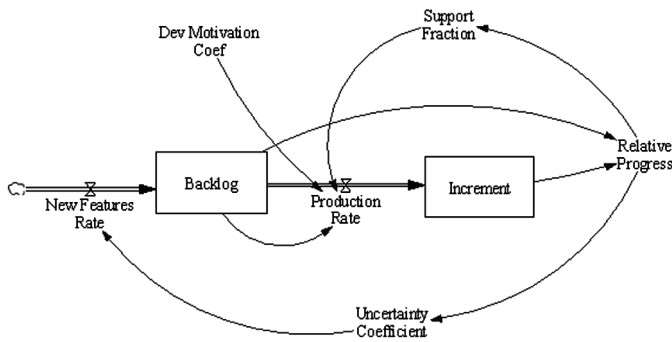


Рис. 12. Модель с учетом изменяющейся продуктивности команды

Проведя повторное моделирование с учетом этих обстоятельств (рис. 13), наблюдаем снижение относительного прогресса (т.е. добавление требований происходит быстрее, чем их реализация) и коэффициента неопределенности (с ростом масштаба программной системы она теряет фокус разработки).

Введем связь разработки программного обеспечения и организационного изменения на базе разработанного продукта (рис. 15). Для описания эффективности организационного изменения будем использовать метрику *Org Change Metric (OCM)*. Модель может быть развита путем введения не-

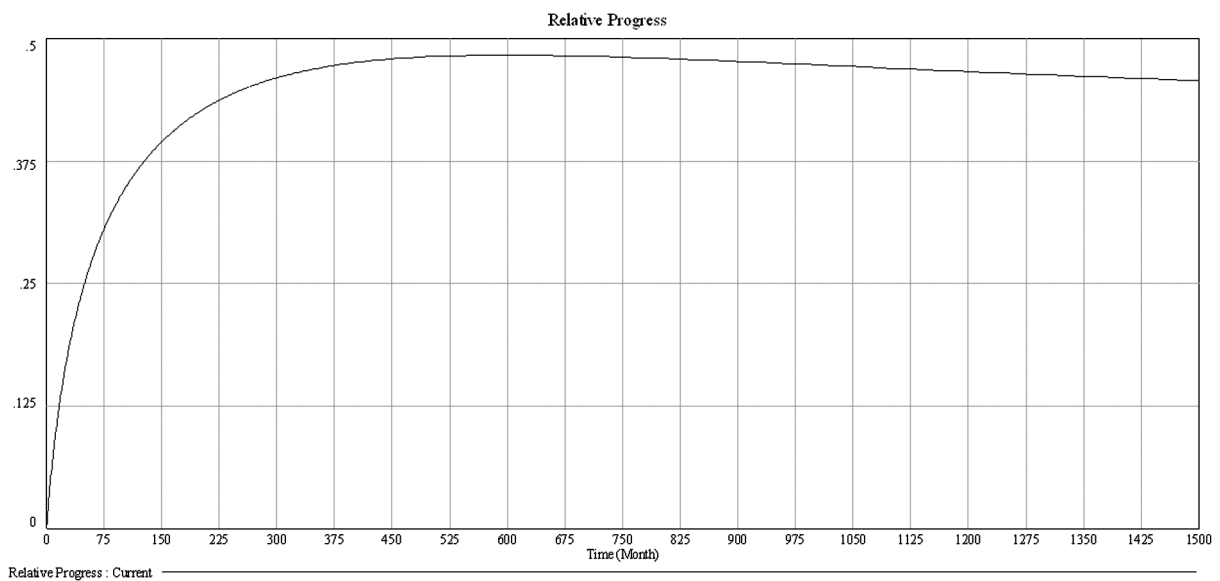


Рис. 13. Результат моделирования (доля выполненных работ)

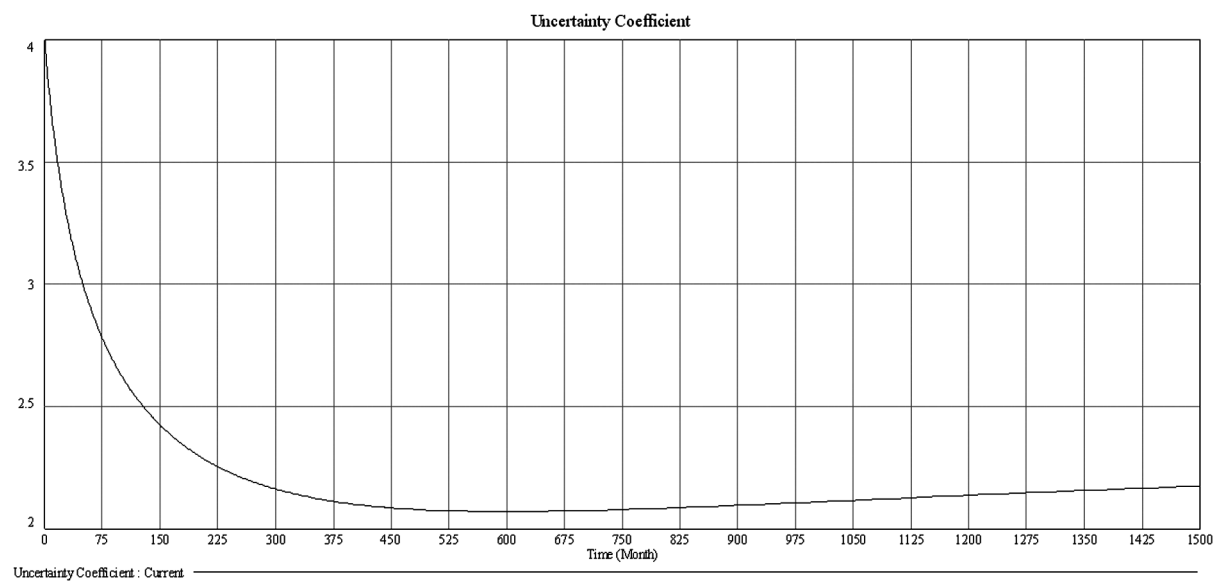


Рис. 14. Результат моделирования (коэффициент неопределенности)

скольких метрик. В данной модели будет использоваться одна интегральная метрика, измеряемая в относительных единицах *сп*.

Темп выполнения организационного изменения (скорость изменения интегральной метрики) — *Org Change Rate (OCR)*. Единица измерения данного параметра — *сп/день*. Дифференциальное уравнение, описывающее организационное изменение, будет выглядеть следующим образом:

$$\frac{dOCM}{dt} = OCR. \tag{9}$$

Переменная *Org Change Rate* при этом связана с переменной *Production Rate (PR)*, поскольку организационное изменение основывается на разработанном продукте. Следующие параметры оказывают влияние на реализацию в организационном изменении возможностей, предоставляемых разработанным программным продуктом:

- *Deployment Delay (DD)* — средняя задержка от реализации функционального требования в программном продукте до реализации организационного изменения, на которое функциональное требование направлено;
- *Feature Efficiency (FE)* — нормирующий коэффициент между объемом реализованного требования (в *сп*) и организационного изменения, которое оно позволяет реализовать при полном (сто процентном) внедрении в операционную деятельность. Данный коэффициент может быть как положительным, так и в некоторых случаях (когда разработанный продукт не может улучшить бизнес-процессы, а только ухудшает их) отрицательным;
- *Implementation Index (II)* — доля функциональных возможностей продукта, внедренная в операционную деятельность.

$$OCR(t) = PR(t - DD) \cdot FE \cdot II. \tag{10}$$

С метрикой организационного изменения также связаны вспомогательные переменные, которые будут использоваться в следующем параграфе (при синтезе устойчивой системы управления *IT*-проектом):

- *Org Change Planned (OCP)* — план реализации организационного изменения, выраженный через планируемое изменение метрики в зависимости от выполнения работ по разработке (в соответствии с модифицированным методом освоенного объема);

- *Org Change Gap (OCG)* — отклонение фактического значения метрики от плана.

$$OGG = OCP - OCM. \tag{11}$$

Relative Gap (RG) — относительное отклонение.

$$RG = \frac{OCG}{OCP}. \tag{12}$$

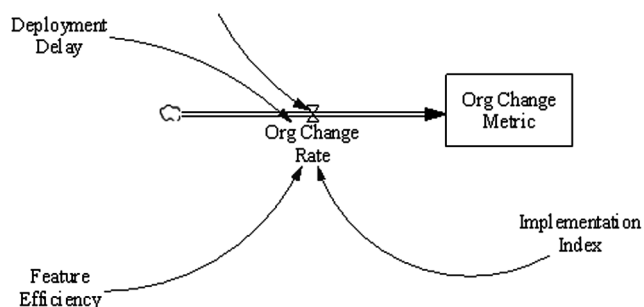


Рис. 15. Связь с метрикой организационного изменения

И на эффективность реализованного функционала в продукте (*Feature Efficiency*) с точки зрения организационного изменения, и на индекс внедрения (*Implementation Index*) оказывает влияние расползание требований вследствие неопределенности, введенное в модель выше.

Механизм снижения эффективности разработанного функционала при расползании требований состоит в том, что продукт теряет целостный образ, соответствие или несоответствие функционала, включаемого в требования, целям проекта становится неочевидным. В условиях, когда отсутствует регулярный процесс сбора метрик проводимого организационного изменения и анализа причин отклонений от планируемых значений, отсутствуют основанные на фактах оценки ценности разработанного функционала и, как следствие, невозможно не только улучшение процессов разработки продукта, но и понимание того, какая часть разработанного функционала не поддерживает цели проекта или даже вредит их реализации.

Scope Creep Coefficient (SCC) — коэффициент расползания требований.

$$SCC = \frac{B + I}{IS}, \tag{13}$$

где *Initial Scope (IS)* — объем требований к продукту *B* в момент начала разработки.

$$FE = (1 - sf \cdot (SCC - 1)) \cdot NFE, \tag{14}$$

где sf — параметр системы управления, выражающий влияние расползания требований на эффективность разработанного функционала.

Nominal Functional Efficiency (NFE) — нормирующий коэффициент, номинальная средняя эффективность, устанавливает связь между реализованной в продукте функциональностью и организационным изменением в условиях, когда расползание требований отсутствует. Единицы измерения коэффициента — cp/sp .

Индекс внедрения *Implementation Index (II)*, в свою очередь, определяется как

$$II = \frac{ii}{SCC}, \tag{15}$$

т.е. чем больше коэффициент расползания требований, тем меньшая доля функциональных возможностей. В действительности эффективно используется в операционной деятельности (рис. 16).

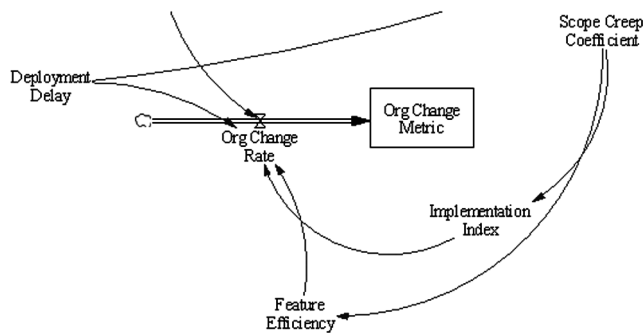


Рис. 16. Эффективность программного продукта и коэффициент внедрения

Далее приведены результаты моделирования (временные диаграммы интегрированной метрики организационного изменения, ее планового значения и отклонения метрики от планового значения), которые свидетельствуют о том, что существуют такие параметры процесса, при которых модель демонстрирует признаки потери устойчивости: интегрированная метрика после наблюдаемого роста в течение времени, примерно соответствующего длительности проекта по базовому плану, начинает снижаться, и этот процесс, если не будет остановлен, может привести к потере даже тех позиций, которые имелись до выполнения проекта (рис. 17–19).

Таким образом, существует потребность во введении изменений в систему управления ИТ-проектом для обеспечения ее устойчивости.

Обеспечение устойчивости системы управления ИТ-проектами

Следующим шагом развития модели, построенной в предыдущем параграфе и иллюстрирующей проблему расползания требований, является введение в модель элементов процесса мониторинга. Этот процесс в соответствии с разработанным алгоритмом включает в себя сбор метрик, аудит реализации возможностей программного продукта в операционной деятельности и соответствия планируемой функциональности продукта целям операционной деятельности. В модели не будут учитываться искажения в получаемых оценках. Искажения в оценках оказывают влияние на соотношение параметров и,

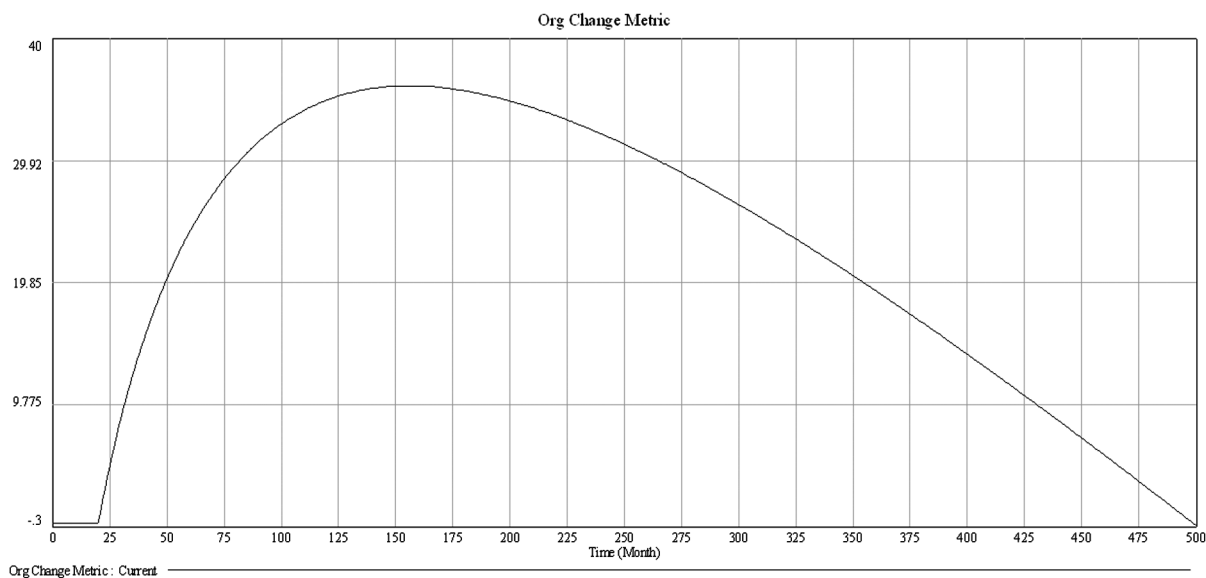


Рис. 17. Результат моделирования (метрика организационного изменения)

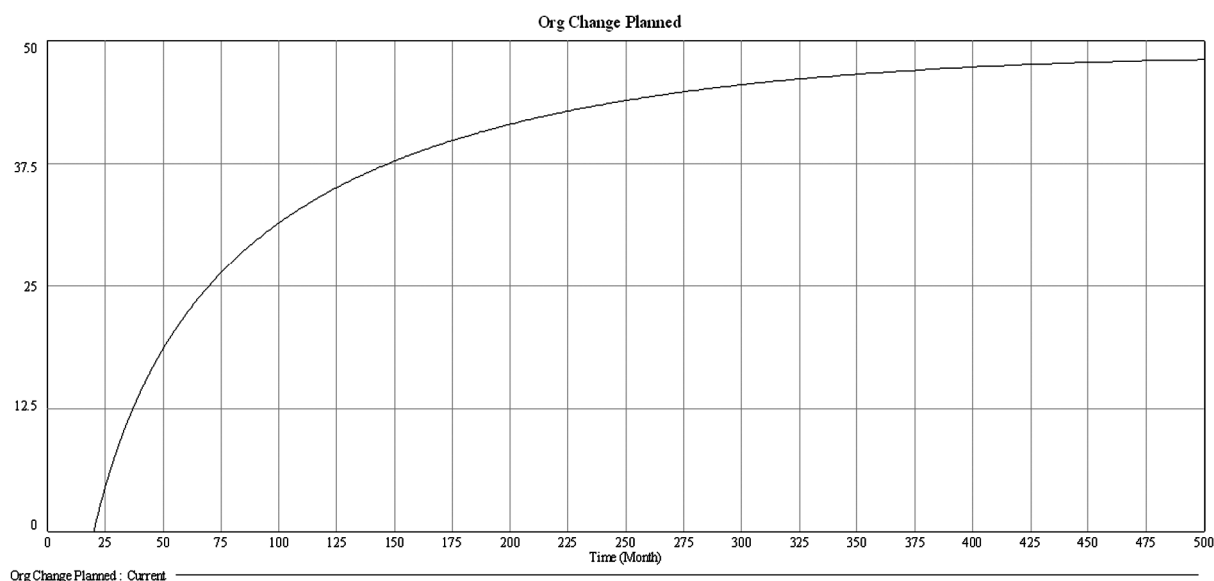


Рис. 18. Результат моделирования (плановое значение метрики)

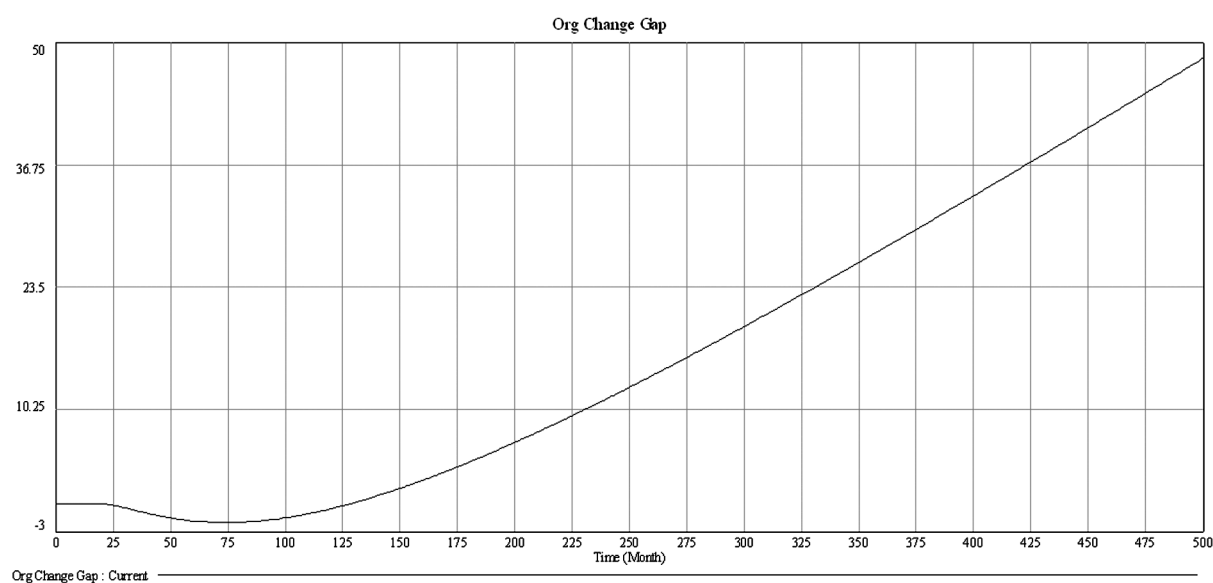


Рис. 19. Результат моделирования (отклонение метрики от плана)

таким образом, на устойчивость системы управления. Но поскольку параметры модели гипотетические, и она ставит целью рассмотрение закономерностей в динамике системы управления *IT*-проектом, то рассмотрение искажений измеренных значений переменных в процессе измерения является для данной модели излишним.

Процессы аудита реализации возможностей продукта в операционной деятельности и соответствия функциональности целям организационного измерения должны иметь в качестве результата пересмотр требований к продукту и их приоритетов и сопро-

вождаться удалением из планируемого содержания продукта требований, относительно которых получено подтверждение того, что они описывают функциональность, не поддерживающую планируемое организационное изменение, или их внедрение в операционную деятельность сложно и должно сопровождаться значительными затратами.

Таким образом, вводится дополнительная переменная состояния и модифицируется уравнение изменения объема требований к продукту (5). Введем переменную *Feature Attrition Rate (FAR)* — темп удаления требований из содержания проекта. При

верно определенных отклонениях метрики реализации организационного изменения и эффективности разработанной функциональности темп удаления требований будет пропорционален этим отклонениям: чем ниже выявленная эффективность требований и чем больше влияние неэффективных требований на организационное изменение, тем больше усилий должно прилагаться к пересмотру запланированных требований к продукту.

Отклонение эффективности разработанной функциональности от планируемой *Feature Efficiency Gap (FEG)* определяется по формуле

$$FEG = 1 - \frac{FE}{NFE}; \tag{16}$$

$$FAR = far \cdot FEG \cdot RG. \tag{17}$$

Уравнение, описывающее изменение планируемого содержания проекта, примет вид:

$$\frac{dB}{dt} = NFR - PR - FAR. \tag{18}$$

Изменения в модели, связанные с управлением требованиями по результатам мониторинга организационного изменения, представлены на рис. 20. Результаты моделирования показывают, что процесс стал сходящимся, и существует точка, в которой реализованы все требования. При этом по-прежнему наблюдается временное ухудшение метрики организационного изменения (рис. 22). Также наблюдается падение эффективности функциональности программного продукта и индекса внедрения фун-

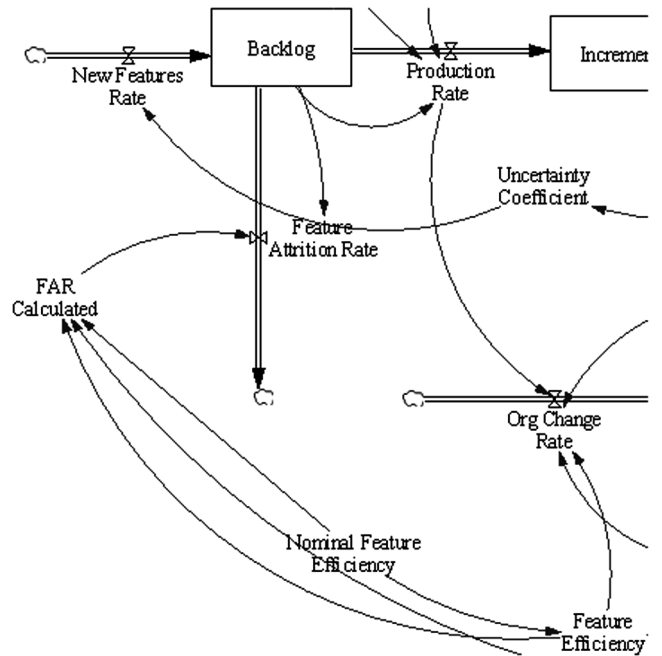


Рис. 20. Моделирование управления требованиями

кциональности продукта в операционную деятельность (рис. 23 и 24). Для решения этих проблем дополним модель двумя однотипными процессами, описывающими развитие компетенций команды проекта в области разработки продукта и навыков сотрудников, задействованных в операционной деятельности, в использовании разработанного программного продукта.

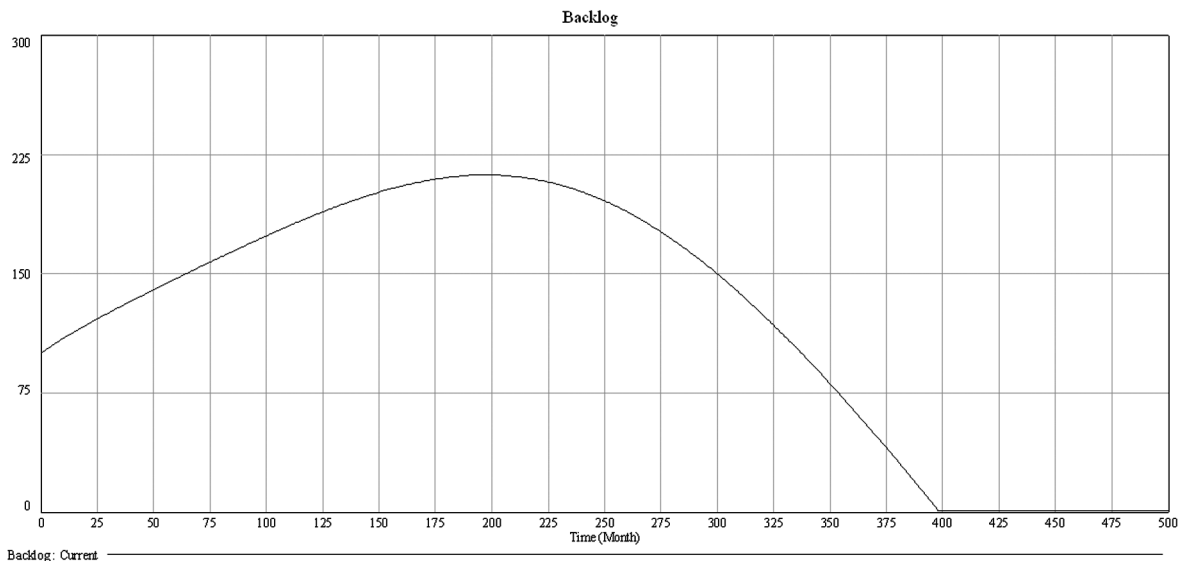


Рис. 21. Результат моделирования (объем требований)

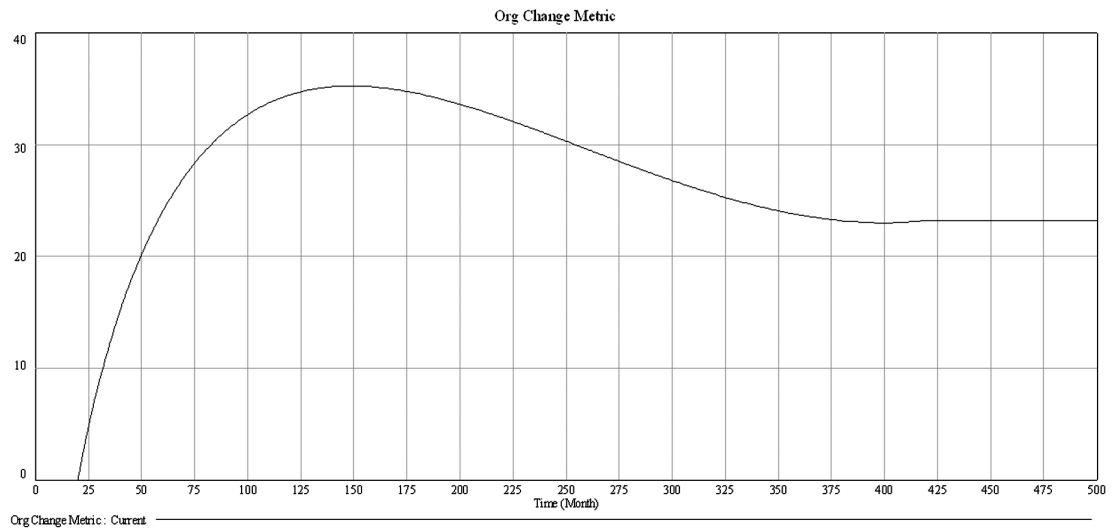


Рис. 22. Результат моделирования (метрика организационного изменения)

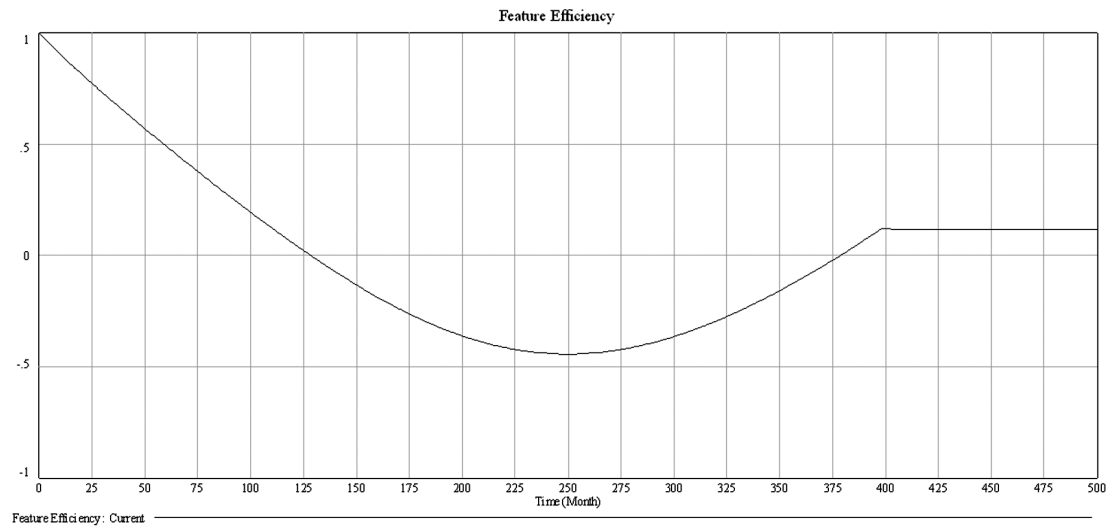


Рис. 23. Результат моделирования (эффективность требований)

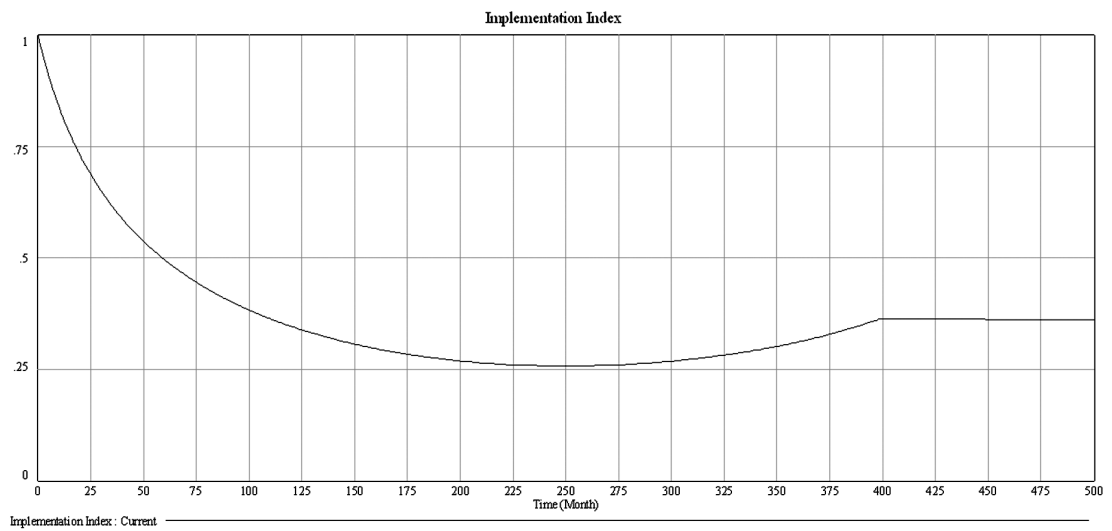


Рис. 24. Результат моделирования (коэффициент внедрения)

Процесс развития навыков разработки представлен на рис. 25, навыков сотрудников, использующих разработанный программный продукт — на рис. 26. В модели используются следующие дополнительные переменные:

- *Dev Skill Coefficient (DSC)* — коэффициент влияния навыков разработки на эффективность функциональности продукта. Навыки наращиваются в процессе обучения в течение разработки, а также при проведении обучения, и имеют свойство устаревать;
- *Dev Training Rate (DTR)* — темп развития компетенций команды в области разработки продукта;
- *Dev Skill Attrition Rate (DAR)* — темп устаревания компетенций команды в области разработки продукта.

Таким образом,

$$\frac{dDSC}{dt} = DTR - DAR; \tag{19}$$

$$FE = (1 - sf \cdot (SCC - 1)) \cdot NFE \cdot DSC. \tag{20}$$

Аналогично *Ops Skill Coefficient (OSC)* — коэффициент влияния навыков персонала, задействованного в операционной деятельности, на индекс внедрения функциональности продукта.

Ops Training Rate (OTR) — темп развития компетенций персонала, задействованного в операционной деятельности.

Ops Skill Attrition Rate (OAR) — темп устаревания компетенций команды в области разработки продукта.

$$\frac{dOSC}{dt} = OTR - OAR; \tag{21}$$

$$II = \frac{ii \cdot OSC}{SCC}. \tag{22}$$

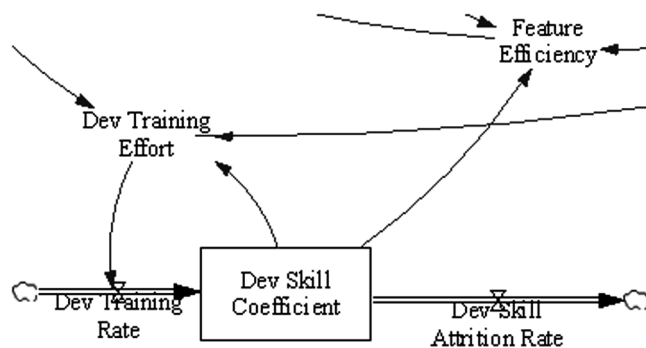


Рис. 25. Процесс развития навыков разработки

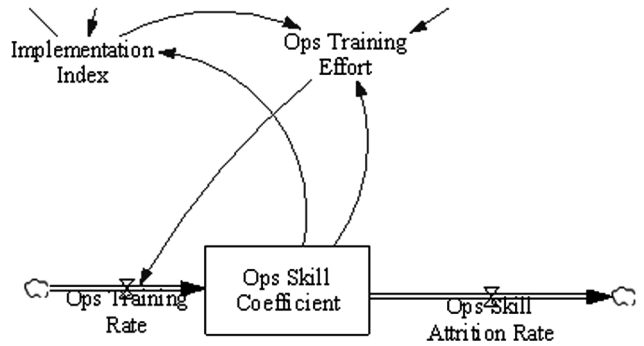


Рис. 26. Процесс развития навыков использования программного продукта

Из результатов моделирования видно, что введение процессов, предполагающих развитие персонала, задействованного как в разработке продукта, так и в операционной деятельности, приводит к стабилизации системы управления. Можно наблюдать, что процесс реализации требований является сходящимся (рис. 30), хотя при заданных параметрах проект выполняется почти с двукратным (рис. 31) превышением трудозатрат. Дальнейшее улучшение процесса с тем, чтобы сократить требуемые трудозатраты на выполнение проекта и его длительность, достигается уже не введением в систему управления проектом новых процессов, обеспечивающих устойчивость, а оптимизацией ее параметров. В данной статье развитие модели будет остановлено на получении модели устойчивой (хотя и неоптимальной) системы управления IT-проектами, так как именно это и являлось целью исследования.

Заключение

В статье подробно рассмотрен вопрос устойчивости системы управления IT-проектами к изменениям содержания, актуальность которого обусловлена тем, что в силу высокой неопределенности изменения содержания являются характерными для таких проектов. Для целей анализа устойчивости IT-проекты рассматриваются в контексте организационных изменений, для осуществления которых они выполняются.

Представленные в работе результаты моделирования показывают, что явление расползания требований, признаваемое исследователями одним из основных факторов, приводящих к увеличению сроков и затрат IT-проектов, также оказывает негативное воздействие на реализацию самого организационного изменения, снижая эффективность функциональности разрабатываемого программного

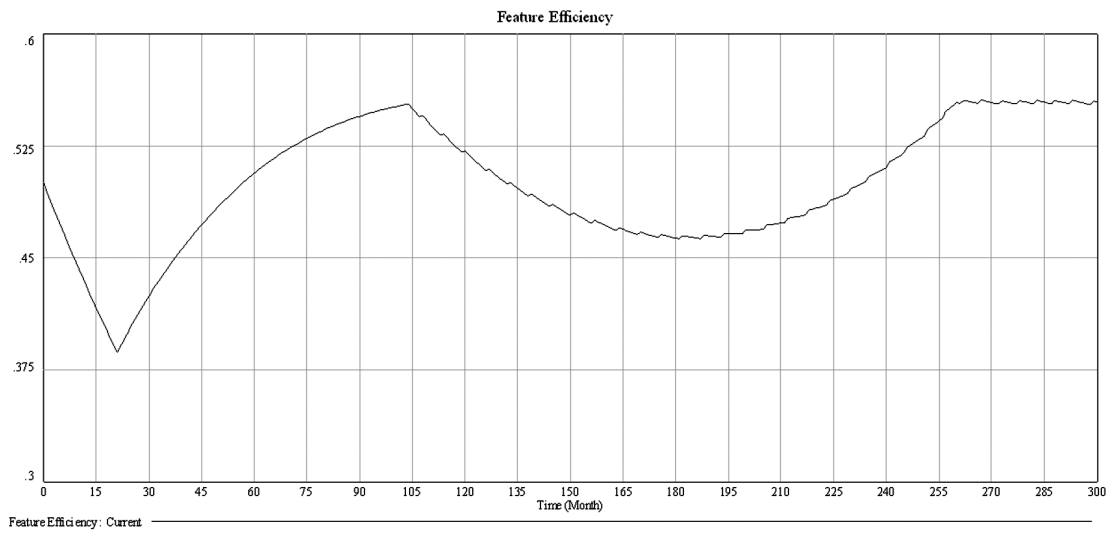


Рис. 27. Результат моделирования (эффективность требований)

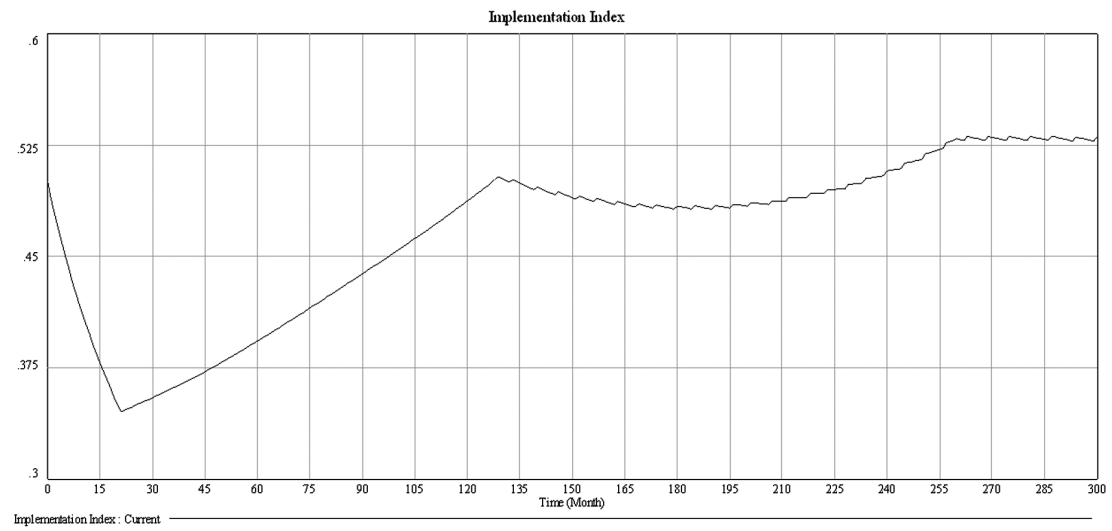


Рис. 28. Результат моделирования (коэффициент внедрения)

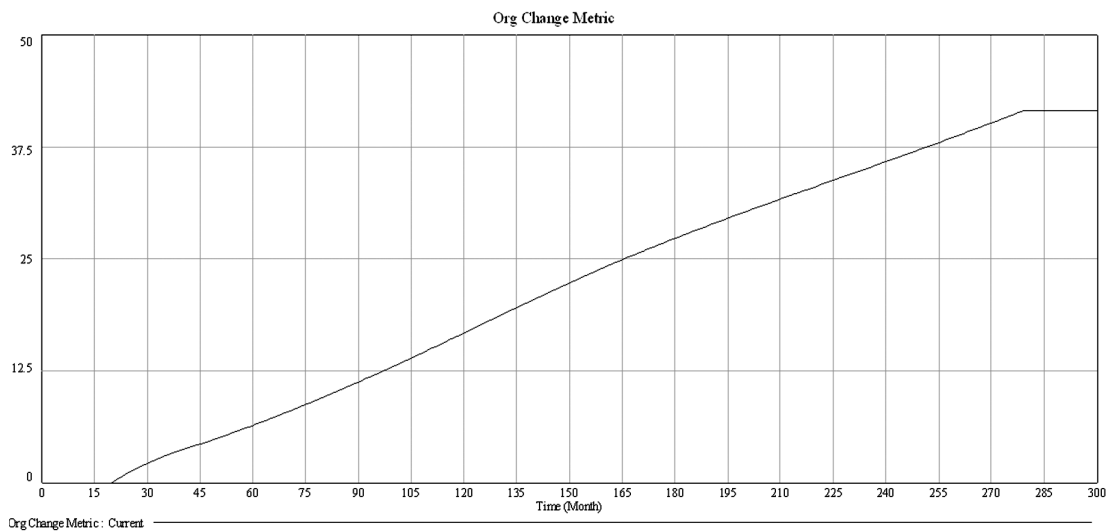


Рис. 29. Результат моделирования (метрика организационного изменения)

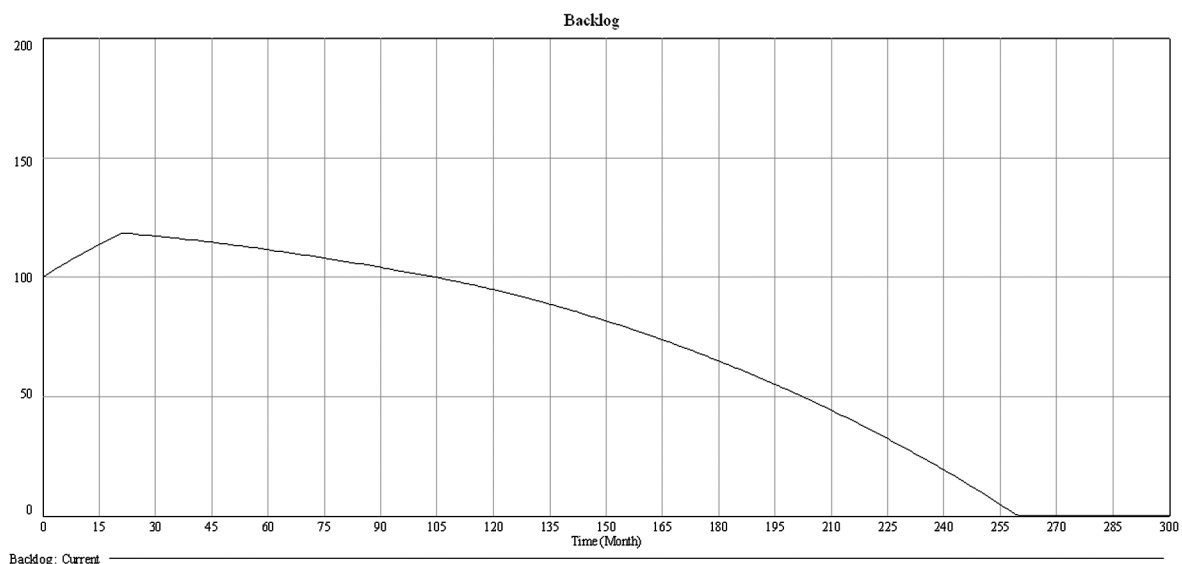


Рис. 30. Результат моделирования (объем требований)

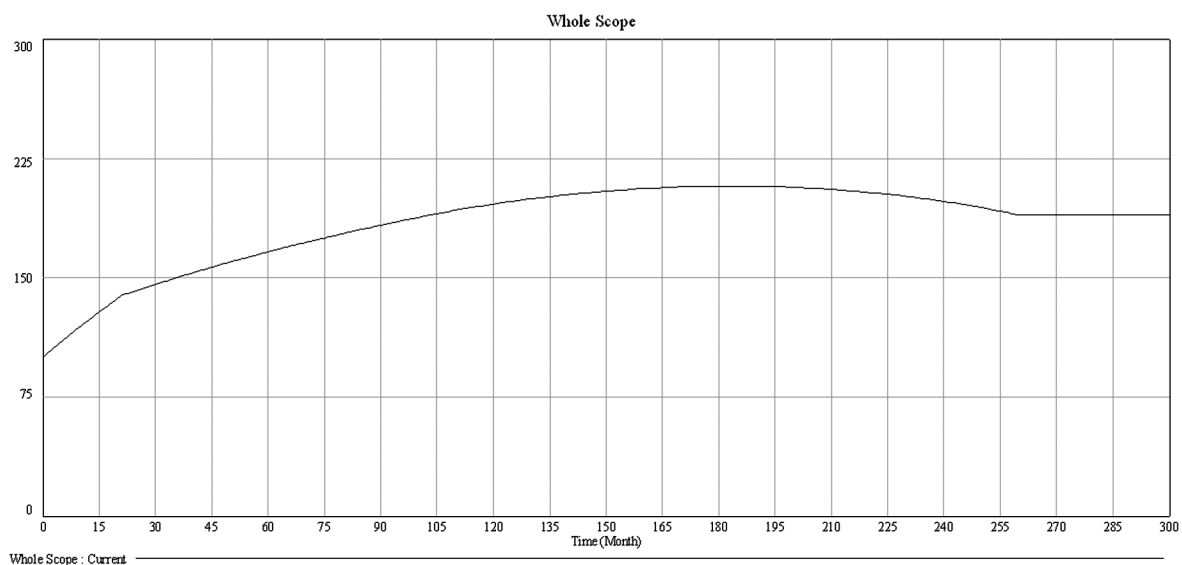


Рис. 31. Результат моделирования (полный объем требований проекта)

обеспечения и уменьшая коэффициент внедрения, выражающий долю функциональности, освоенную персоналом компании и успешно применяющуюся в операционной деятельности.

В работе разработаны инструменты обеспечения устойчивости ИТ-проектов, основывающиеся на введении в систему управления дополнительной обратной связи для отслеживания реализации связанного с выполняемым проектом организационного изменения. В качестве основы для разработанного метода мониторинга был использован метод освоенного объема, дополненный отслеживанием коэффициента эффективности осуществления

запланированного организационного изменения. Предложены мероприятия, которые следует проводить в случае выявления отклонения реализации организационного изменения от плана, и позволяющие выявить причины возникшего отклонения, а также направить усилия на их устранение.

Повторное моделирование с учетом разработанного метода мониторинга показало, что система управления ИТ-проектом приобрела признаки устойчивой системы, т.е. процесс выполнения проектом стал сходящимся, а негативное влияние расползания требований на ход организационного изменения было устранено.

Литература

1. *Abdel-Hamid T.K.* The dynamics of software development project management: an integrative system dynamics perspective, 1984.
2. *Ahmadi S.A.A.* et al. A study on the relationship between organizational structure and organizational agility: A case study of insurance firm // *Manag. Sci. Lett.* 2012. Т. 2. № 8. С. 2777–2788.
3. *Boehm B.W.* Software Engineering Economics. Englewood Cliffs, NJ: Prentice-Hall, 1981.
4. *Brehm L., Markus M.L.* The divided software life cycle of ERP packages // *Proceedings of 1st Global Information Technology Management (GITM) World Conference*, 2000. P. 43–46.
5. *Demirkan H., Goul M.* Service oriented technology management to improve organizational agility // *Int. J. Innov. Technol. Manag.* 2009. Т. 6. № 01. P. 1–16.
6. *Eisenhardt K.M., Tabrizi B.N.* Accelerating adaptive processes: Product innovation in the global computer industry // *Adm. Sci. Q.* 1995. С. 84–110.
7. *Hannan M.T., Freeman J.* Structural inertia, organizational change // *Am. Sociol. Rev.* 1984. Т. 49. № 2. P. 149–164.
8. *Herroelen W., Leus R.* Robust and reactive project scheduling: a review and classification of procedures // *Int. J. Prod. Res.* 2004a. Т. 42. № 8. P. 1599–1620.
9. *Herroelen W., Leus R.* The construction of stable project baseline schedules // *Eur. J. Oper. Res.* 2004b. Т. 156. № 3. P. 550–565.
10. *Herroelen W., Leus R.* Project scheduling under uncertainty: Survey and research potentials // *Eur. J. Oper. Res.* 2005. Т. 165. № 2. P. 289–306.
11. *Hersch M.* Benefit-based earned value management system // 2012.
12. ITSMF UK. An introductory overview of ITIL. London: TSO, 2011.
13. *Jones C.* Strategies for managing requirements creep // *Computer.* 1996. № 6. P. 92.
14. *Kirchmer M., Gutiérrez F., Laengle S.* Process mining for organizational agility // *Ind. Manag. Plaines.* 2010. Т. 52. № 1. P. 19.
15. *Knapp D.* ITSM Process Design Guide. Fort Lauderdale, FL: J. Ross Publishing Inc., 2010. 255 p.
16. *Leus R., Herroelen W.* Stability and resource allocation in project planning // *IIE Trans.* 2004. Т. 36. № 7. P. 667–682.
17. *Lofi C.* Continuous GQM: An automated framework for the goal-question-metric paradigm // 2005.
18. *Madachy R.J.* Software process dynamics: John Wiley & Sons, 2008. 601 p.
19. *Nejatian M., Zarei M.H.* Moving Towards Organizational Agility: Are We Improving in the Right Direction? // *Glob. J. Flex. Syst. Manag.* 2013. Т. 14. № 4. P. 241–253.
20. *O'Toole J.* Change resisted. Thirty-three hypotheses why // *Organization Change: A Comprehensive Reader*: Jossey-Bass, 1995. P. 13.
21. PMI. Software Extension to the PMBOK Guide, Fifth Edition.: Project Management Institute, 2013. Vol. 5. 268 p.
22. *Portougal V., Sundaram D.* Business processes: operational solutions for SAP implementation. Hershey, PA: IRM Press, 2006. 329 p.
23. *Serra C.E.M., Kunc M.* Benefits Realisation Management and its influence on project success and on the execution of business strategies // *Int. J. Proj. Manag.* 2015. Т. 33. № 1. С. 53–66.
24. *Sulaiman T., Barton B., Blackburn T.* Agile EVM-earned value management in Scrum Projects // *Agile Conference*, 2006: IEEE, 2006. P. 10–16.
25. *Swartz S.M.* Managerial perceptions of project stability // *Proj. Manag. J.* 2008. Т. 39. № 4. P. 17–32.
26. *Thakurta R.* Impact of Scope Creep on Software Project Quality // *Vilakshan XIMB J. Manag.* 2013. Т. 10. № 1.
27. *Uhl A., Gollenia L.* A Handbook of Business Transformation Management Methodology: Ashgate Publishing Ltd, 2012. 345 p.
28. *Vonder S. Van de, Demeulemeester E., Herroelen W.* Proactive heuristic procedures for robust project scheduling: An experimental analysis // *Eur. J. Oper. Res.* 2008. Т. 189. № 3. P. 723–733.
29. *Аксенов Е.* Аутсорсинг: 10 заповедей и 21 инструмент [Текст] / Е. Аксенов, И. Альтшулер. — СПб.: Питер, 2009. — 464 с.
30. *Брукс П.* Метрики для управления ИТ-услугами [Текст] / П. Брукс. — М.: Альпина Бизнес Букс, 2008. — 283 с.
31. *Бушуев С.Д.* Методологии управления проектами на моделях класса «движущие силы — сопротивление» [Текст] / С.Д. Бушуев, Р.Ф. Ярошенко // *Управління розвитком складних систем.* — 2010. — № 2. — С. 11–14.
32. *Лич Л.* Вовремя и в рамках бюджета: Управление проектами по методу критической цепи [Текст] / Л. Лич. — М.: Альпина Паблишерз, 2010.
33. *Пащенко Д.С.* О факторах, тормозящих развитие российских ИТ-компаний в фазе зрелости [Текст] / Д.С. Пащенко // *Менеджмент и бизнес-администрирование.* — 2013. — С. 123–128.
34. *Ципес Г.Л.* Проекты организационных изменений в крупных компаниях: методы оценки и принятия решений [Текст] / Г.Л. Ципес, А.В. Кузьмищев // *Управление проектами и программами.* — 2014. — № 3. — С. 6–14.
35. *Ципес Г.Л.* Проекты и управление проектами в современной компании [Текст] / Г.Л. Ципес, А.С. Товб. — М.: Олимп-Бизнес, 2009. — 480 с.

References

1. Abdel-Hamid T.K. The dynamics of software development project management: an integrative system dynamics perspective, 1984.
2. Ahmadi S.A.A. A study on the relationship between organizational structure and organizational agility: A case study of insurance firm. *Manag. Sci. Lett.* 2012, V. 2, I. 8, pp. 2777–2788.
3. Boehm B.W. Software engineering economics, 1981.
4. Brehm L., Markus M.L. The divided software life cycle of ERP packages. *Proceedings of 1st Global Information Technology Management (GITM) World Conference.* , 2000, pp. 43–46.
5. Demirkan H., Goul M. Service oriented technology management to improve organizational agility. *Int. J. Innov. Technol. Manag.* 2009, V. 6, I. 01, pp. 1–16.
6. Eisenhardt K.M., Tabrizi B.N. Accelerating adaptive processes: Product innovation in the global computer industry. *Adm. Sci. Q.* 1995, pp. 84–110.
7. Hannan M.T., Freeman J. Structural inertia, organizational change. *Am. Sociol. Rev.* 1984, V. 49, I. 2, pp. 149–164.
8. Herroelen W., Leus R. Robust and reactive project scheduling: a review and classification of procedures. *Int. J. Prod. Res.* 2004a. V. 42, I. 8, pp. 1599–1620.

9. Herroelen W., Leus R. The construction of stable project baseline schedules. *Eur. J. Oper. Res.* 2004b. V. 156, I. 3, pp. 550–565.
10. Herroelen W., Leus R. Project scheduling under uncertainty: Survey and research potentials. *Eur. J. Oper. Res.* 2005, V. 165, I. 2, pp. 289–306.
11. Hersch M. Benefit-based earned value management system, 2012.
12. itSMF UK. An introductory overview of ITIL. London: TSO, 2011.
13. Jones C. Strategies for managing requirements creep. *Computer.* 1996, I. 6, pp. 92.
14. Kirchmer M., Gutiérrez F., Laengle S. Process mining for organizational agility. *Ind. Manag. Plaines.* 2010, V. 52, I. 1, pp. 19.
15. Knapp D. ITSM Process Design Guide. Fort Lauderdale, FL: J. Ross Publishing Inc., 2010. 255 s.
16. Leus R., Herroelen W. Stability and resource allocation in project planning // *IIE Trans.* 2004, V. 36, I. 7, pp. 667–682.
17. Lofi C. Continuous GQM: An automated framework for the goal-question-metric paradigm, 2005.
18. Madachy R.J. *Software process dynamics.* : John Wiley & Sons, 2008. 601 s.
19. Nejatian M., Zarei M.H. Moving Towards Organizational Agility: Are We Improving in the Right Direction? *Glob. J. Flex. Syst. Manag.* 2013, V. 14, I. 4, pp. 241–253.
20. O'Toole J. *Change resisted. Thirty-three hypotheses why. Organization Change: A Comprehensive Reader.*: Jossey-Bass, 1995, pp. 13.
21. PMI. *Software Extension to the PMBOK Guide, Fifth Edition.* : Project Management Institute, 2013, I. 5. 268 p.
22. Portugal V., Sundaram D. *Business processes: operational solutions for SAP implementation.* Hershey, PA: IRM Press Publ., 2006. 329 p.
23. Serra C.E.M., Kunc M. Benefits Realisation Management and its influence on project success and on the execution of business strategies. *Int. J. Proj. Manag.* 2015, V. 33, I. 1, pp. 53–66.
24. Sulaiman T., Barton B., Blackburn T. AgileEVM-earned value management in Scrum Projects. *Agile Conference, 2006.*: IEEE, 2006, pp. 10.
25. Swartz S.M. Managerial perceptions of project stability. *Proj. Manag. J.* 2008, V. 39, I. 4, pp. 17–32.
26. Thakurta R. Impact of Scope Creep on Software Project Quality. *Vilakshan XIMB J. Manag.* 2013, V. 10, I. 1.
27. Uhl A., Gollenia L. *A Handbook of Business Transformation Management Methodology.*: Ashgate Publishing Ltd, 2012. 345 p.
28. Vonder S. Van de, Demeulemeester E., Herroelen W. Proactive heuristic procedures for robust project scheduling: An experimental analysis. *Eur. J. Oper. Res.* 2008, V. 189, I. 3, pp. 723–733.
29. Aksenov E., Al'tshuler I. Outsourcing: 10 zapovedey i 21 instrument [Outsourcing: the 10 commandments and 21 tool]. St. Petersburg, Piter Publ., 2009. 464 p.
30. Bruks P. *Metriki dlya upravleniya IT-uslugami [Service Management].* Moscow, Al'pina Biznes Buks Publ., 2008. 283 p.
31. Bushuev S.D., Yaroshenko R.F. Metodologii upravleniya proektami na modelyakh klassa «dvizhushchie sily - so-protivlenie» [Methodologies for project management class models "driving forces - resistance"]. *Upravlinnyia Rozvitkom Skladnikh Sistem [Management Rozvitkom Folding System].* 2010, I. 2, pp. 11–14.
32. Lich L. Vovremya i v ramkakh byudzheta: Upravlenie proektami po metodu kriticheskoy tsepi [Leach on time and within budget: Project Management Critical Chain method]. Moscow, Al'pina Pabliher Publ., 2010.
33. Pashchenko D.S. O faktorakh, tormozyashchikh razvitie rossiyskikh IT-kompaniy v faze zrelosti [Factors hindering the development of Russian IT-companies in the maturity phase]. *Menedzhment I Biznes-Administrirovanie [Management and Business Administration].* 2013, pp. 123–128.
34. Tsipes G.L., Kuz'mishchev A.V. Proekty organizatsionnykh izmeneniy v krupnykh kompaniyakh: metody otsenki i prinyatiya resheniy [Projects of organizational change in large companies: the methods of assessment and decision-making]. *Upravlenie Proektami I Programmami [Management of projects and programs].* 2014, I. 3, pp. 6–14.
35. Tsipes G.L., Tovb A.S. Proekty i upravlenie proektami v sovremennoy kompanii [Projects and project management in modern companies]. Moscow, ZAO «Olimp-Biznes» Publ., 2009. 480 p.